User's Manual
TK-78K0R/KG3

Date published: June 2008
Rev. 1.1
© TESSERA TECHNOLOGY INC. 2007-2008
Printed in Japan

Windows and Windows XP are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

・The information in this document is current as of February, 2007. The information is subject to change without notice. For actual design-in, refer to the latest publications of the product data sheets.
・No part of this document may be copied or reproduced in any form or by any means without prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
・NEC Electronics does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
・Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
・While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
・NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific". The recommended applications of NEC Electronics products depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product before using it.

"Standard": Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots.

"Special": Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support).

"Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support, etc.

The quality grade of NEC Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers need to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)
(1) "NEC Electronics" as used in this statement means NEC Electronics Corporation and also includes its majority-owned subsidiaries.
(2) "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above).

CAUTION

・Do not give any physical damage to this equipment such as dropping

・Do not superimpose voltage to this equipment.

・Do not use this equipment with the temperature below 0℃ or over 40℃.

・Make sure the USB cables are properly connected.

・Do not bend or stretch the USB cables.

・Keep this equipment away from water.

・Take extra care to electric shock.
・This equipment should be handled like a CMOS semiconductor device. The user must take all precautions to avoid build-up of static electricity while working with this equipment.
・All test and measurement tool including the workbench must be grounded.
・The user/operator must be grounded using the wrist strap.
・The connectors and/or device pins should not be touched with bare hands.

Contents

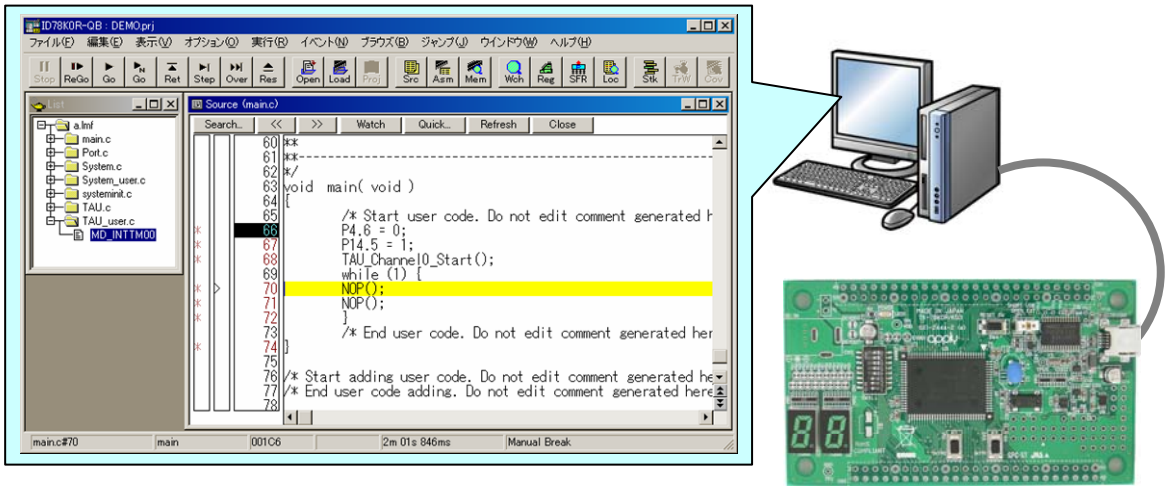
CHAPTER 1 PREPARATION....8
1.1 Development Tools / Software....9
1.2 Installation of Development Tools....10
1.2.1 Installation Package....10
1.2.2 Installation of Development Tools....10
1.3 Installation of USB Driver....16
1.3.1 Installation on Windows XP....17
1.3.2 Installation on Windows 2000....20
1.3.3 Completion of USB Driver Installation....23
1.4 Sample Programs....24
1.4.1 Preparation of Sample Programs....24
1.4.2 Overview of Sample Programs....27
CHAPTER 2 EXPERIENCES....28
2.1 Start PM+....30
2.2 What is PM+....31
2.3 Load Workspace (project)....33
2.4 Set Linker Options....35
2.4.1 "Output1" Tab....35
2.4.2 "Output2" Tab....37
2.5 Set Compiler Options....38
2.5.1 "Extend" Tab....38
2.5.2 "Startup Routine" Tab....39
2.6 Create Load Module Files....40
2.7 Check Debugger Settings....42
2.8 Check Board Settings....44
2.9 Start Debugger (ID78K0R-QB)....45
2.10 Run Programs....48
2.11 Stop Programs....50
2.12 Close Debugger (ID78K0R-QB)....51
2.13 Quit PM+....52
CHAPTER 3 HARDWARE SPECIFICATIONS....53
3.1 Layout of hardware functions....54
3.2 Layout of solder-short pad....54
3.3 Hardware Functions....55
3.3.1 SW1....55
3.3.2 SW2....55
3.3.3 SW3....55
3.3.4 SW4 (RESET SW)....56
3.3.5 LED1 (POWER)....56
3.3.6 JP1....56

3.3.7 U1,U2(7seg LED) 57
3.4 Universal Area 57
3.5 Solder-short pad label 58
With using the solder-short pad to cut the circuit, users can customize the circuit. 58
3.6 Operation by external power supply 59
CHAPTER 4 TROUBLESHOOTING 60
4.1 If you cannot find USB driver when you connect PC to the kit 60
4.2 Error when you start the debugger 60
4.2.1 "Can not communicate with Emulator..." (F0100 or A0109) 61
4.2.2 "Incorrect ID Code." (Ff603) 62
4.2.3 "The on-chip debug function had been disabled in the device." (F0c79) 62
4.2.4 "Disabling the on-chip debug function is prohibited." (F0c33) 62
CHAPTER 5 OTHER INFORMATION 63
5.1 Create a new workspace 64
5.2 Register additional source file 69
5.3 Debugger tips 71
5.3.1 Change display of buttons 71
5.3.2 Display source list and function list 71
5.3.3 Set/delete breakpoints 72
5.3.4 Display global variables 73
5.3.5 Display global variables while programs are running 74
5.3.6 Display local variables 75
5.3.7 Display memory and SFR contents 75
5.3.8 Erase microcontroller built-in flash memory 76
5.4 Circuit diagram 78

Introduction

TK-78K0R/KG3 is the evaluation kit for development using "78K0R/Kx3", NEC Electronics 16bit all flash microcontroller.
The user only needs to install the development tools and USB driver, and connect the host machine with the target board to start the code development, build, monitoring the output, and debugging code.
(This demonstration kit uses the on-chip debug feature from the microcontroller itself, without emulator connection)

Configuration for Debugging



By loading the sample programs, you can confirm running the board.

Input/output for operation check use
・7Seg LED
・Push switch

Overview This manual consists of the following contents.

Read chapter 1 and 2 first for installing the development tools and using the sample programs.

Read chapter 3-5 for customizing the sample programs and the hardware.

Chapter 1: Preparations
Install the development tools

Chapter 2: Experiences
Experience the basic operations of integrated development environment (PM+) and integrated debugger (ID78K0R-QB) with using sample programs.

Chapter 3: Hardware Specifications
Explain the hardware of TK-78K0R/KG3

Chapter 4: Troubleshooting
Describe how to solve troubles you may face, such as errors when starting the integrated debugger (ID78K0R-QB)

Chapter 5: Other Information
Introduce other information, such as how to create a new workspace (project) on integrated development environment (PM+), how to register additional source file, and some useful tips of the integrated debugger. The circuit diagrams of demonstration kit are included in this chapter.

Reader This manual is intended for development engineers who wish to become familiar with the development tools for the 78K0R.
It is assumed that the readers have been familiar with basics of microcontrollers, C and Assembler languages, and the Windows™ operating system.

Purpose This manual is intended to give users an understanding of the features, hardware configurations, development tools for the 78K0R.

CHAPTER 1　Preparation

This chapter describes following topics:

- Overview and installation of development tools
- Installation of development tools
- Overview and preparation of sample programs

Users can experience the development flow such as coding, build, debugging, and test, by using the development tools bundled with TK-78K0R/KG3.

1.1 Development Tools / Software

- Device file DF781188 V3.00

  A device file contains device specific information. So, users need a device file to use the development tools.

- Integrated Development Environment (IDE) PM+ V6.30

  The IDE works on Windows operation system.
  Users can develop a system efficiently by using the editor with idea processor function, compiler, and debugger.

- C Compiler CC78K0R　W1.20 (code size limited version)

  C compiler for the 78K0R microcontrollers. The object code size is limited to 64 Kbyte.
  This compiles C code for 78K0R and ANSI-C code program into assembler code.
  This produces object code and linker.

- Assembler RA78K0R　W1.10 （code size limited version）

  Assembler for the 78K0R microcontrollers. The object code size is limited to 64 Kbyte.
  This convert the assembler code for 78K0R into object program. The object program will be used for debugger.

- 78K0R Integrated Debugger ID78K0R-QB V3.40

  This is the tool for debugging the object program generated by C compiler and assembler. The debugger enables to do C source level debugging. With the debugger, you can debug the code easily and efficiently by refering and changing variables, using step-in debuging function, and so on.

- Starter Kit USB Driver

  USB driver for the USB connection with the TK-78K0R/KG3 and PC.

- Sample program

  Demonstration program using the 7seg-LED and push switch.

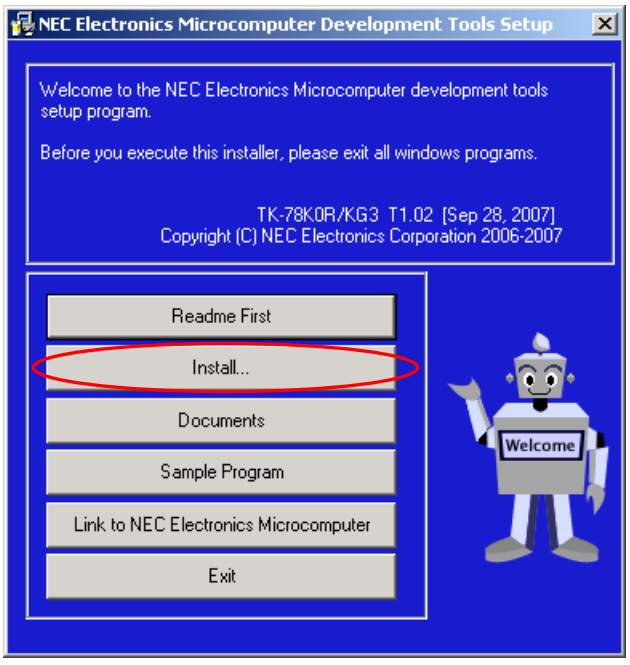
1.2 Installation of Development Tools

1.2.1 Installation Package

The attached CD-ROM includes the development tools and documentations. Users can use the installer to install those development tools and documentations.

1.2.2 Installation of Development Tools

① Please insert the CD-ROM in the drive. The installer will show up automatically. If it does not start automatically, please initiate it by double clicking the SETUP.EXE.



<1> Readme First

The contents of the CD-ROM, and some notes are available.

Please read it at first.

<2> Install…

Click "Install" to start installation of development tools.

For details, please refer to the next section.

<3> Documents

Manuals of development tools and the evaluation kit are available in PDF files.

When this button is clicked, the WWW browser will start. Adobe® Acrobat® Reader is available in the CD-ROM.

<4> Sample Program

Click this button to start the WWW browser for the sample program and the tutorial.

<5> Link to NEC Electronics Microcontrollers

Click this button to start the WWW browser display the link to the NEC Electronics Microcontroller web site (http://www.necel.com/micro/index_e.html)

The NEC Electronics Microcontroller web page provides with the latest product/tool information and FAQs.

<6> Exit

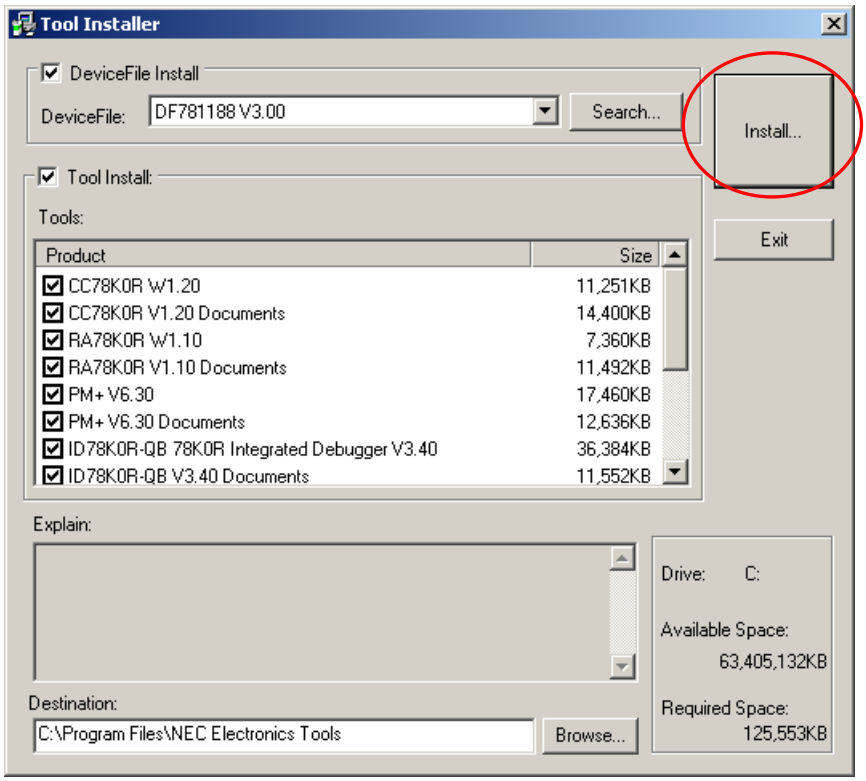
Terminate the setup.

② Click the "Install"

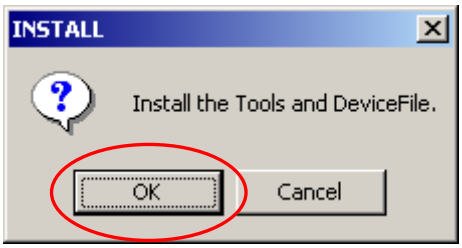
③ "Tool Installer" dialog box is opened.
Select products that you need to install.
(as default, all the products that you need to use the TK-78K0R/KG3 are selected.)
"Explain" area displays an explanation of the selected product.
To change the installation destination, click Browse··· .

When all the settings are completed, click Install··· .

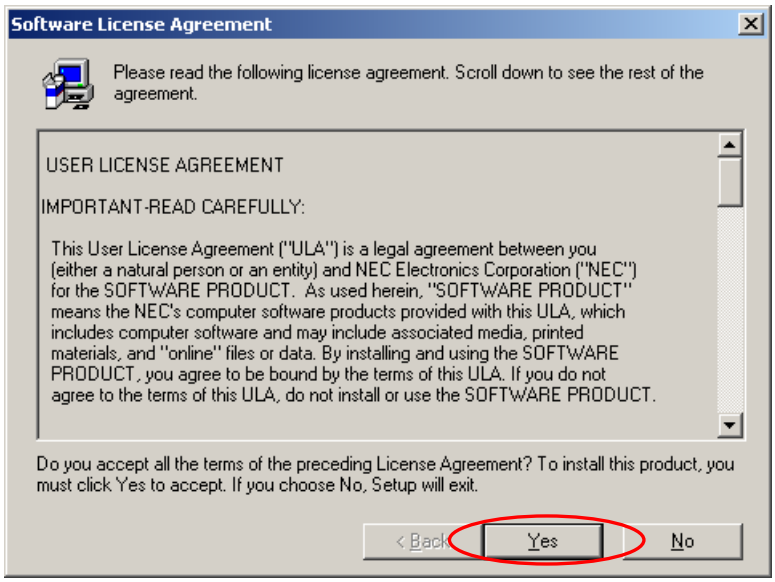
* In this document, it is assumed that users install the programs under "NEC Electronics Tools" directory (default installation directory). Users can find the tools by selecting "Start Menu" -> "Programs" -> "NEC Electronics Tools".



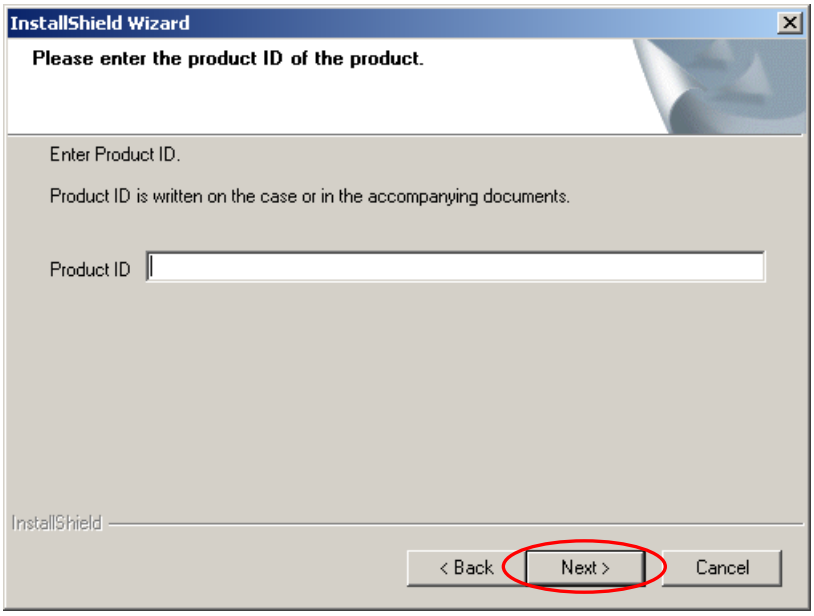
④ Click OK when "Install" comfirmation dialog box is opened.



⑤ Read "software license agreement" and click [Yes] for continuing the installation.
To stop the installation, click [No] .

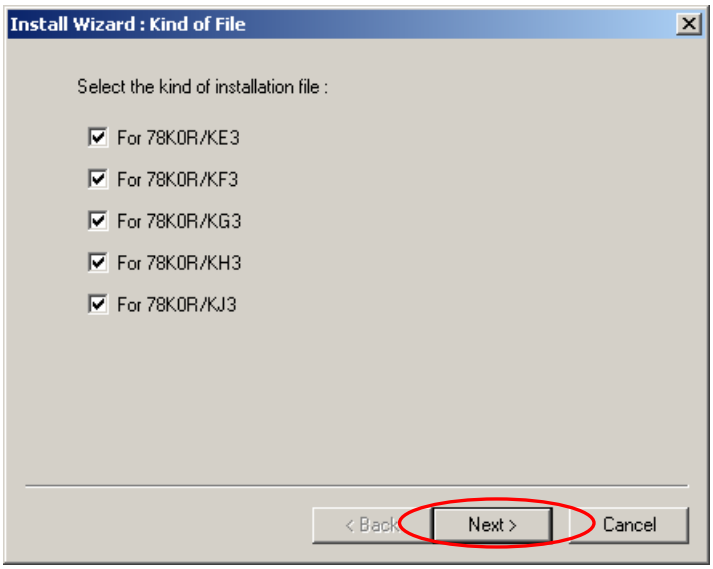


⑥ Enter the product ID, and click [Next] .
＊ The product ID is available on the other sheet.

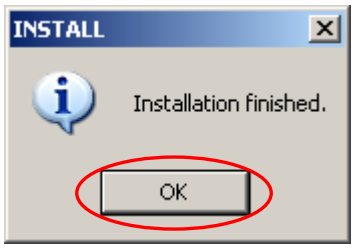


⑦ It starts copying the files.

⑧ Click Next when "Select Files" installation wizard dialog opened.



⑨ When the installation is completed, the following dialog opens.
Click OK .



⑩ "NEC Electronics Starter Kit Virtual UART" USB driver must be installed on PC before you connect to TK-78K0R/KG3.

Install the USB driver by referring "1.3 Installation of USB Driver".

Notes on the installation authority

To install this tool in Windows 2000 or XP, the authority of an administrator is necessary. Therefore, please login as an administrator.

Notes on the install-directory

Please do not use 2-byte characters, such as umlaut in the directory name, where the product is to be installed.

Note on the version of Windows

If the language of the Windows is not English, a file transfer error during installation might be observed. In this case, please abort the installation in the language, and re-install it in an English version of Windows.

The identical problem may be observed, if a language other than English is specified as the system language in the "Regional Settings Properties" tab.

Limitation

Assembler RA78K0R and C compiler CC78K0R limit the object size to 64 Kbyte.

1.3 Installation of USB Driver

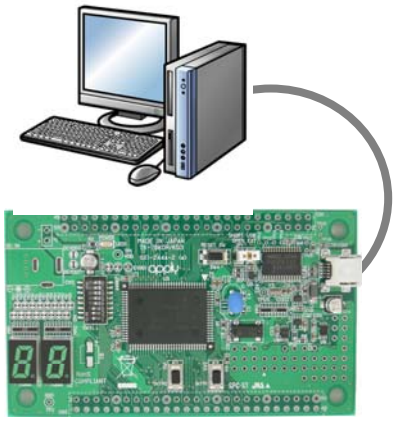
"NEC Electronics Starter Kit Virtual UART" USB driver must be installed on PC before you start using the TK-78K0R/KG3.
Please, follow the instruction below to install the driver.

"Starter Kit USB Driver" must be installed on the PC. If not, please refer to "1.2 Installation of Development Tools" to install the driver first.

CAUTION:

Do not use a USB hub for connecting TK-78K0R/KG3.

First, connect the TK-78K0R/KG3 to PC with USB.



Depending on the version of Windows OS, the installation will be differed. Please check your Windows version, and follow the instructions

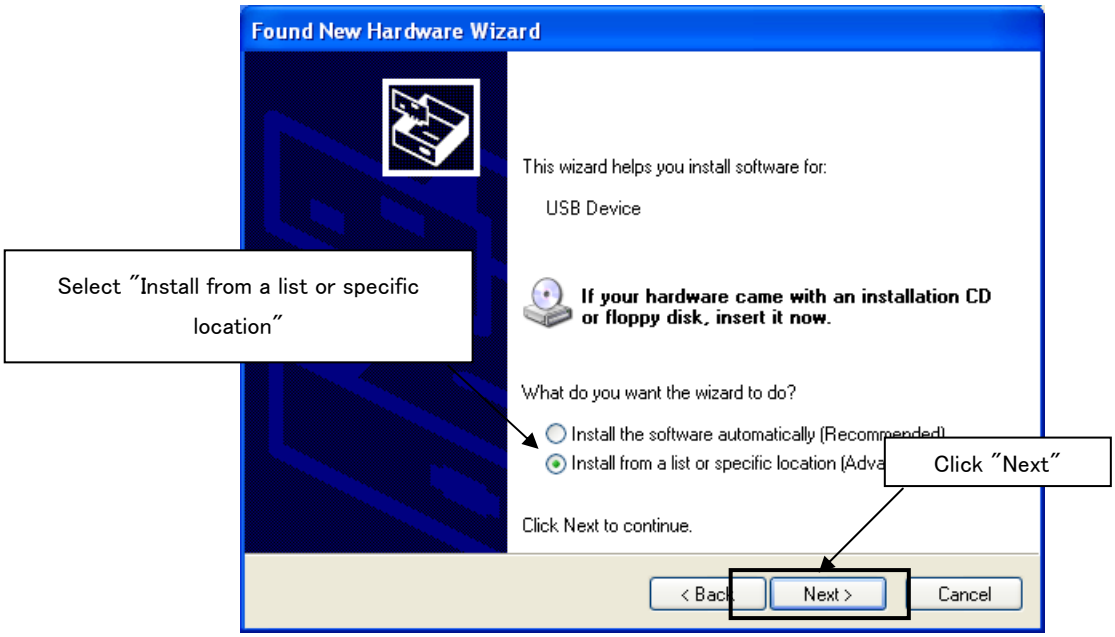
- Windows XP -> "1.3.1 Installation on Windows XP"
- Windows 2000 -> "1.3.2 Installation on Windows 2000"

After the installation, go to "1.3.3 Completion of USB Driver Installation"

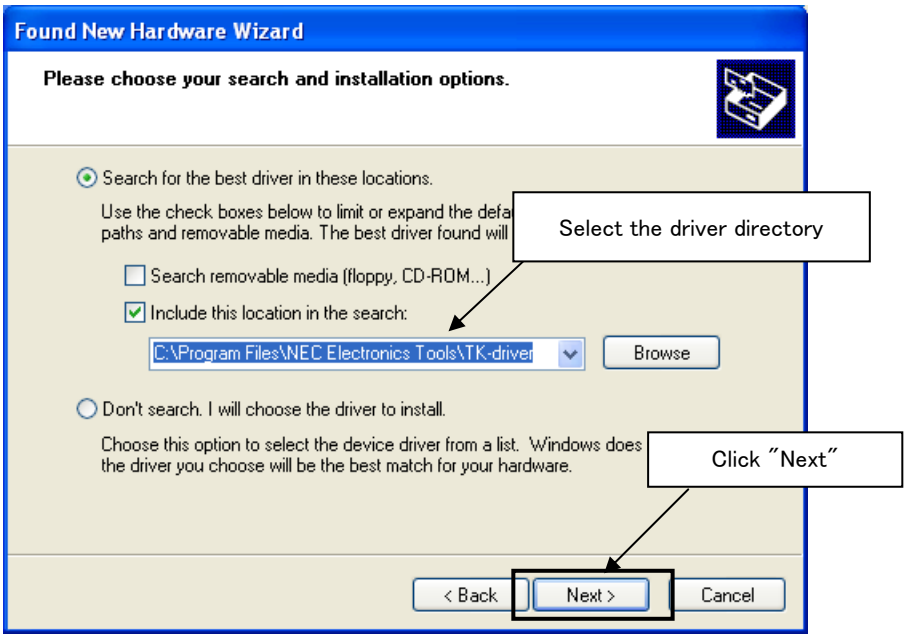
1.3.1 Installation on Windows XP

1. Once the TK-78K0R/KG3 is connected with USB, the "Found New Hardware Wizard" will be started.
   Select "No, not this time" and click Next > .

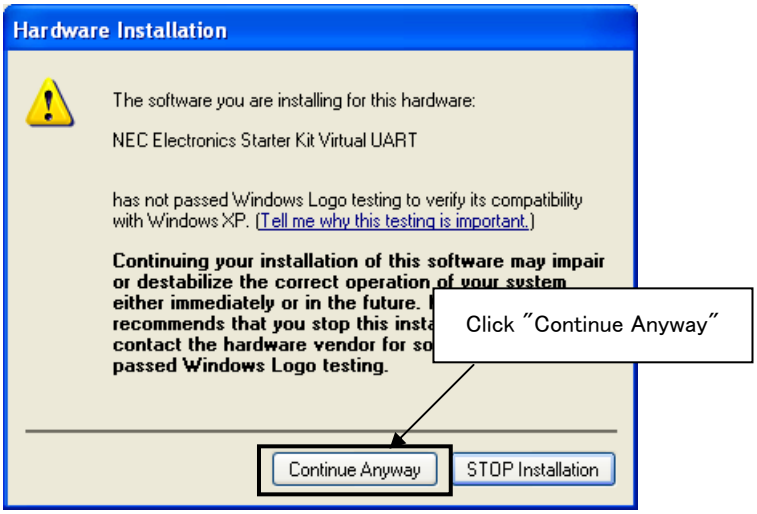
2. Select "Install from a list or specific location" and click Next > .



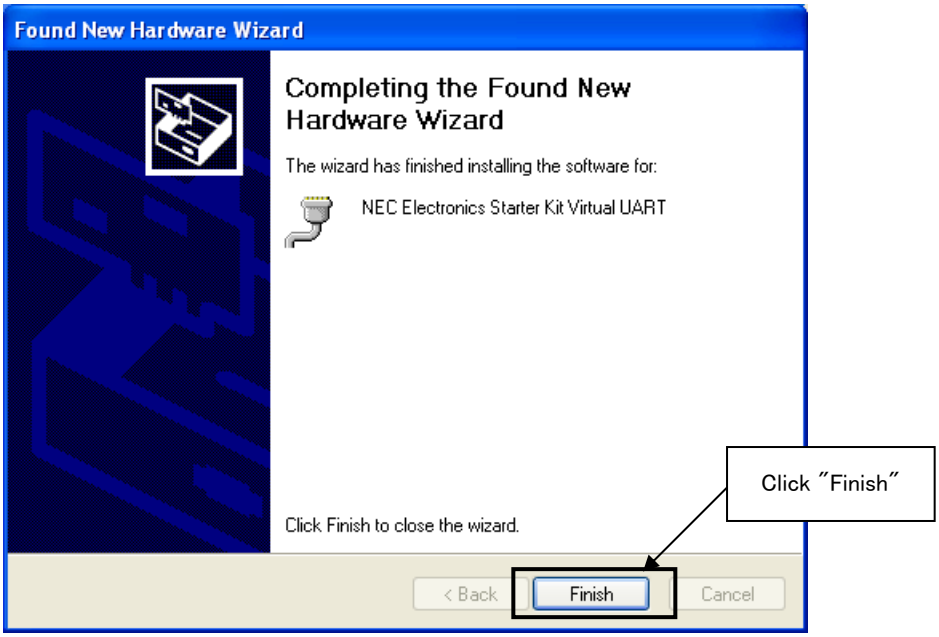
3. Select "Search for the best driver in these locations.", check "Include this location in the search:", and then click "Browse..." to select the driver directory path. The path should be "C:¥Program Files¥NEC Electronics Tools¥TK-driver" as default installation. If the installation directory is not default, then select "TK-driver" under the installation directory.
Click [Next >] .



4. If the following dialog is opened, click [Continue Anyway] .



6. The installation of "NEC Electronics Starter Kit Virtual UART" driver is completed.
   Click Finish .

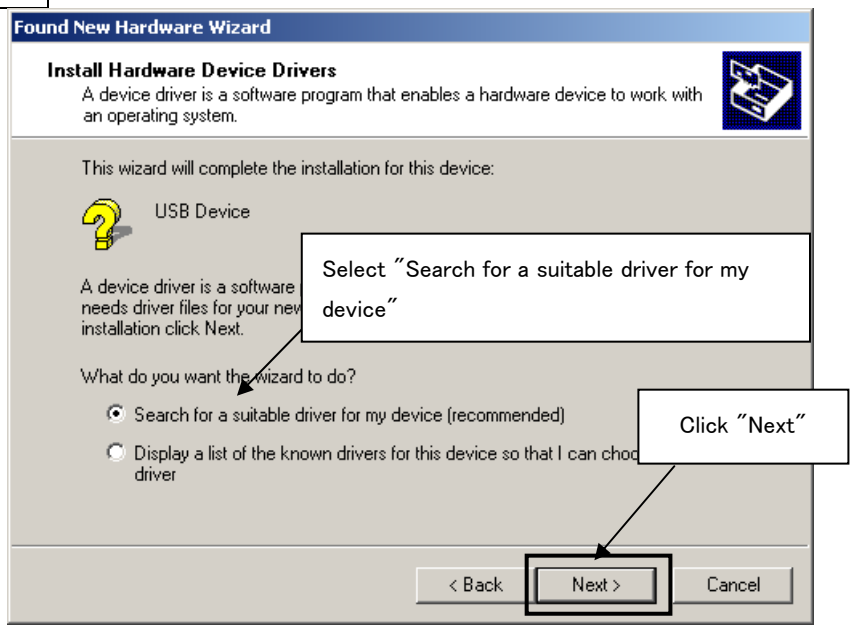


7. Go to "1.3.3 Completion of USB Driver Installation".

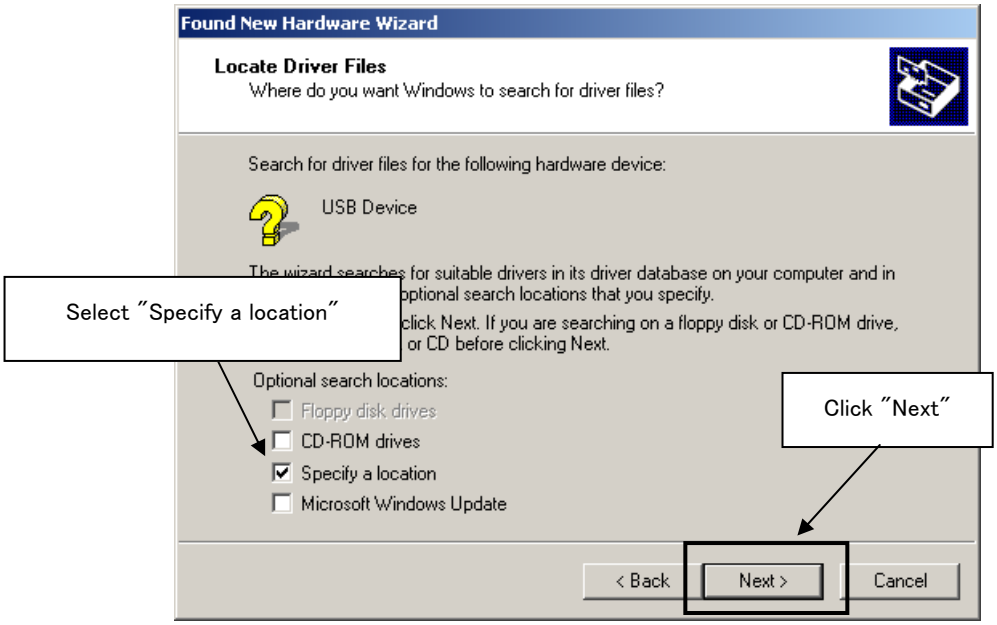
1.3.2 Installation on Windows 2000

1. Once the TK-78K0R/KG3 is connected with USB, the "Found New Hardware Wizard" will be started.
   Select "No, not this time" and click [Next >] .

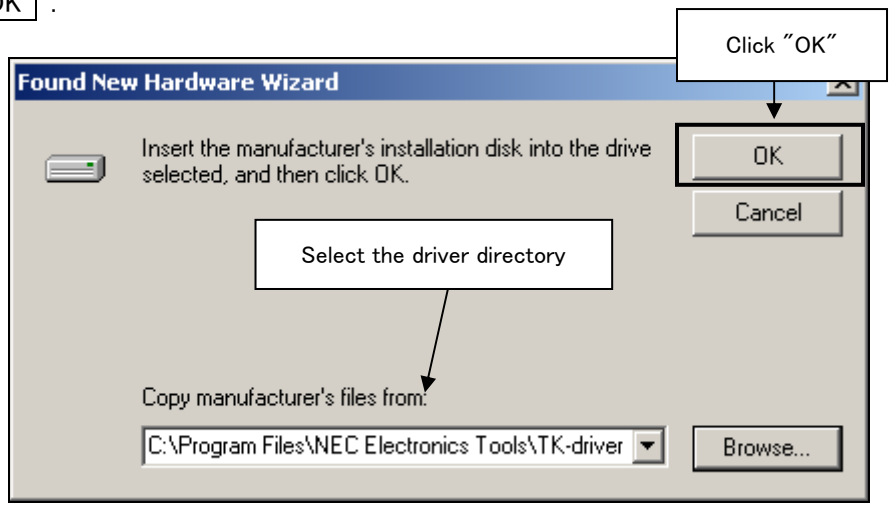
2. Select "Search for a suitable driver for my device".
   Click [Next >] .



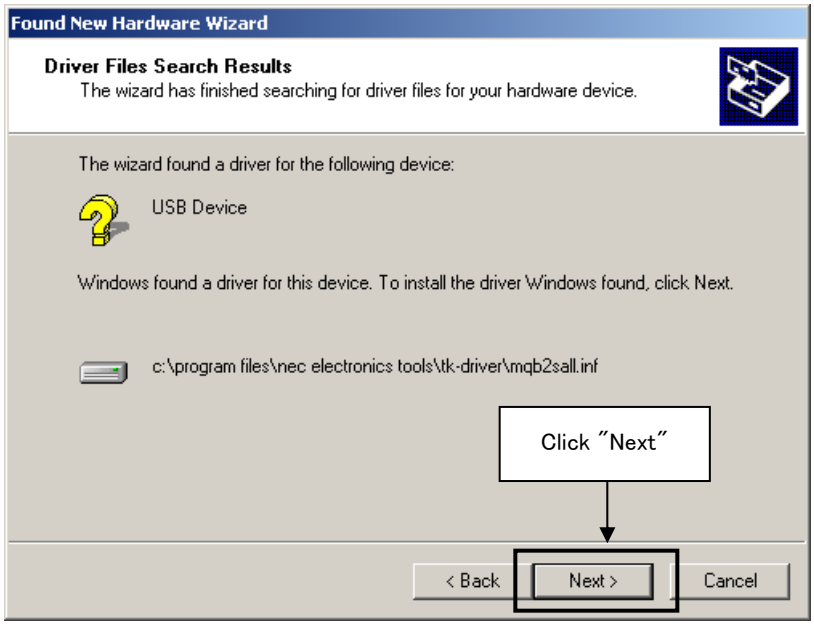
3. Select "Specify a location".
   Click [Next >] .



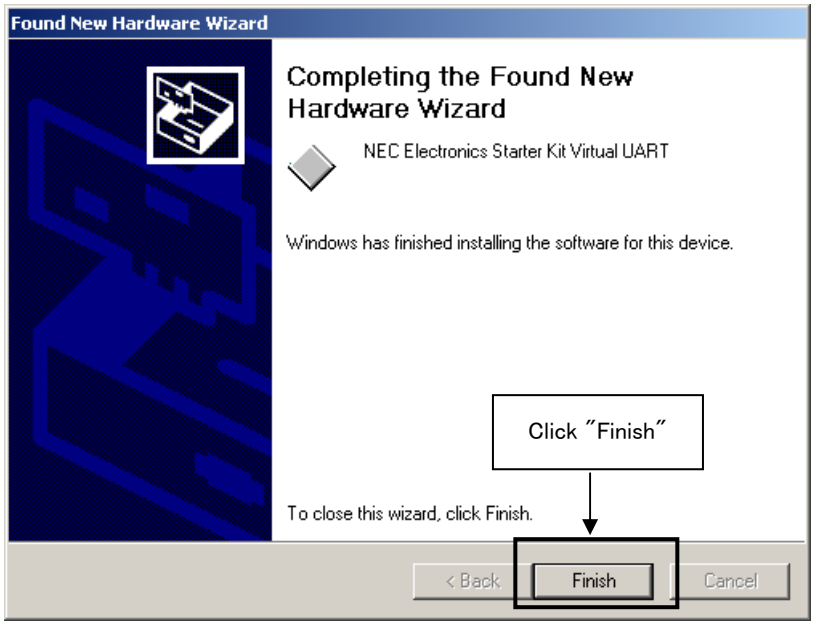
4. Select the driver directory path. The path should be "C:¥Program Files¥NEC Electronics Tools¥TK-driver" as default installation.
   If the installation directory is not default, then select "TK-driver" under the installation directory.
   Click [OK] .



5. Click Next > .



6. The installation of "NEC Electronics Starter Kit Virtual UART" driver is completed. Click Finish .



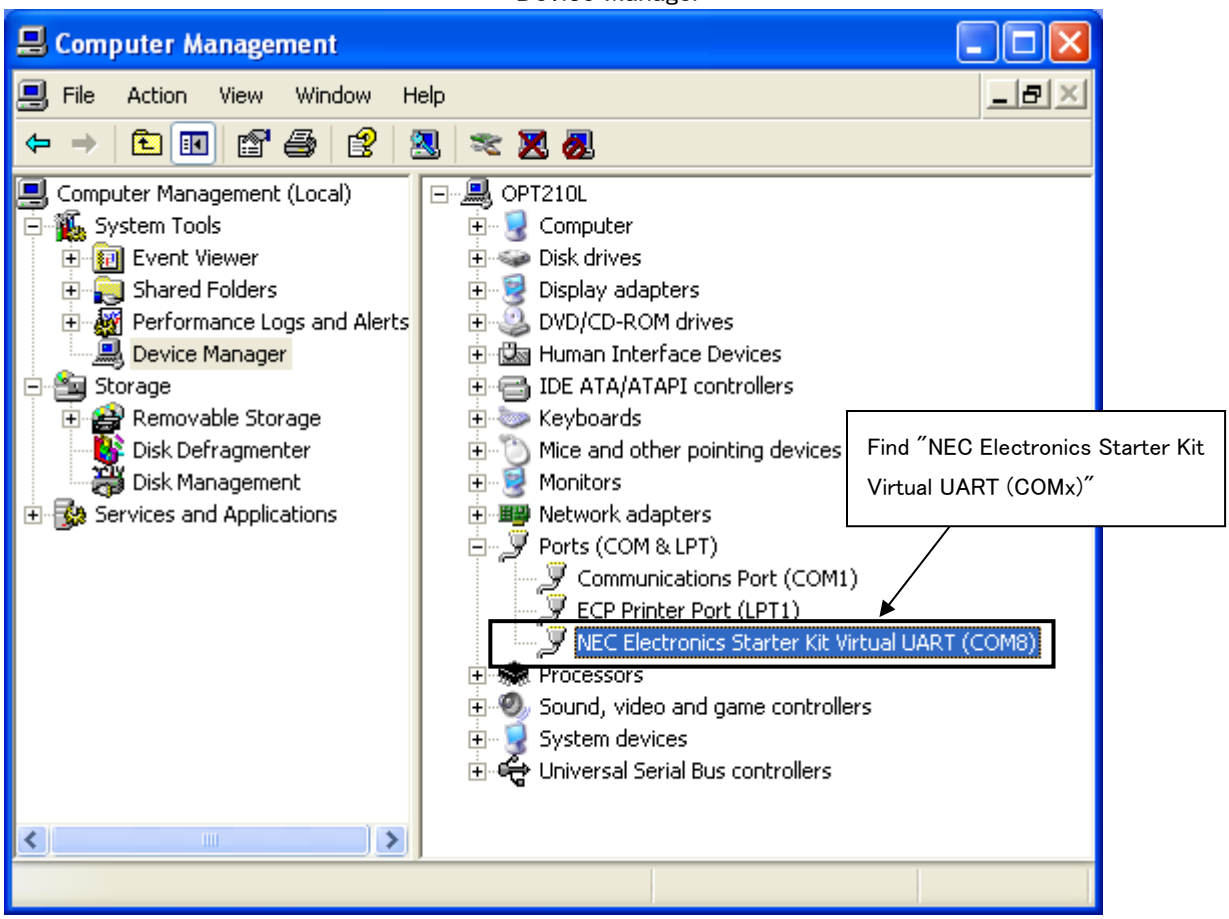
7. Go to "1.3.3 Completion of USB Driver Installation".

1.3.3 Completion of USB Driver Installation

Confirm the USB driver is installed on PC.

Start "Device Manager", and find "NEC Electronics Starter Kit Virtual UART" (without "?" mark) under the "Ports (COM & LPT)".

Device Manager



The screen above shows that the COM port number is "COM8". If ID78K0R-QB is not in use, you can use this port number for connecting TK-78K0R/KG3.
When you change the USB port connection, the COM port number will be changed as well.

CAUTION

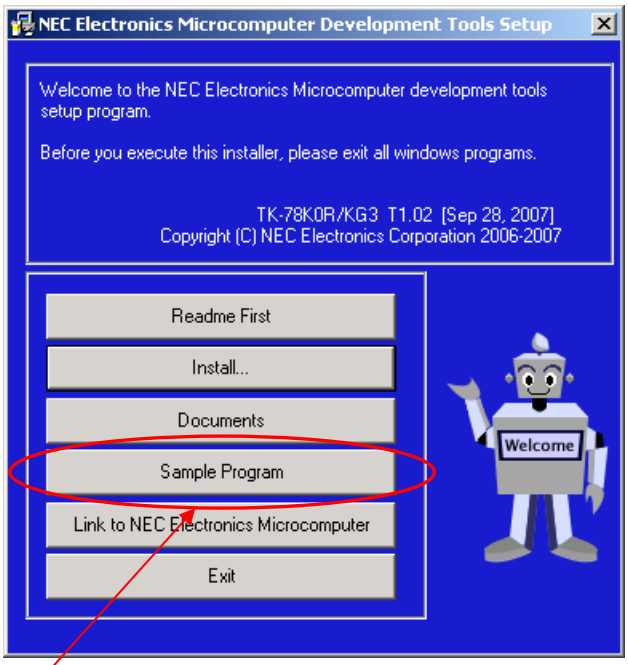
・Do not do "Hardware Modification Scan" when you communicate with the target device.

1.4 Sample Programs

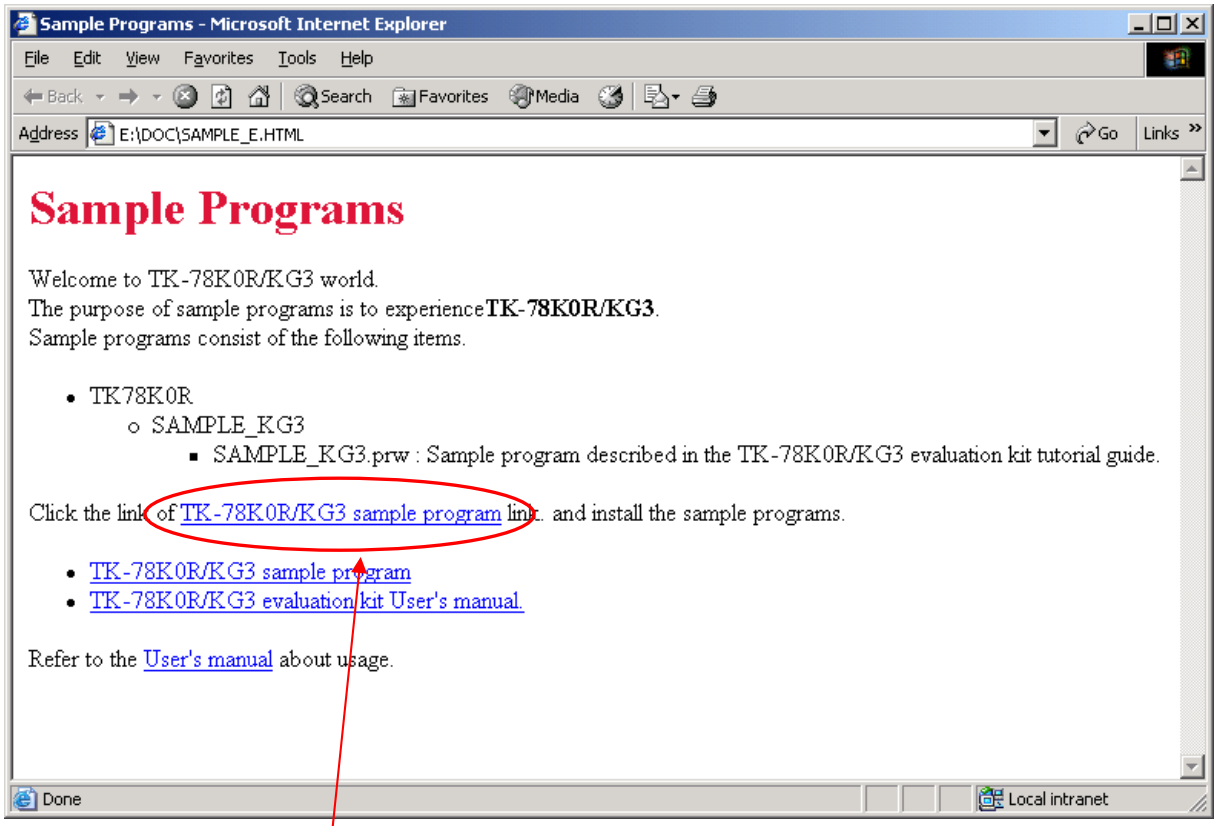
This section explains the overview and preparation of sample programs.

1.4.1 Preparation of Sample Programs

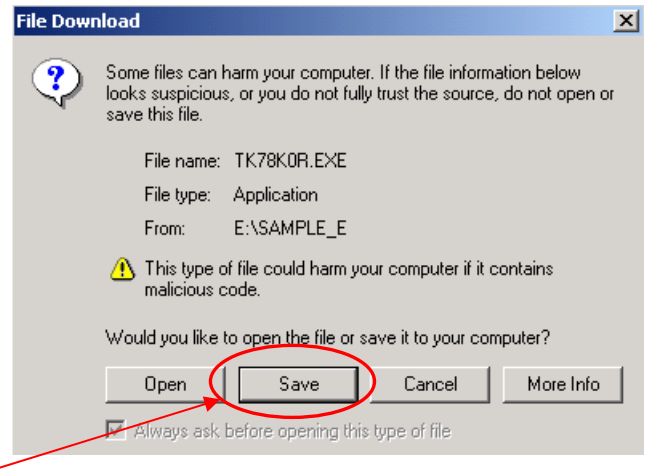
① Insert the CD-ROM disk in the CD-ROM drive of your PC. The [NEC Electronics Microprocessor Development Tools Setup] screen automatically appears.(if this screen does not appear automatically, start setup.exe from Explorer. etc.)



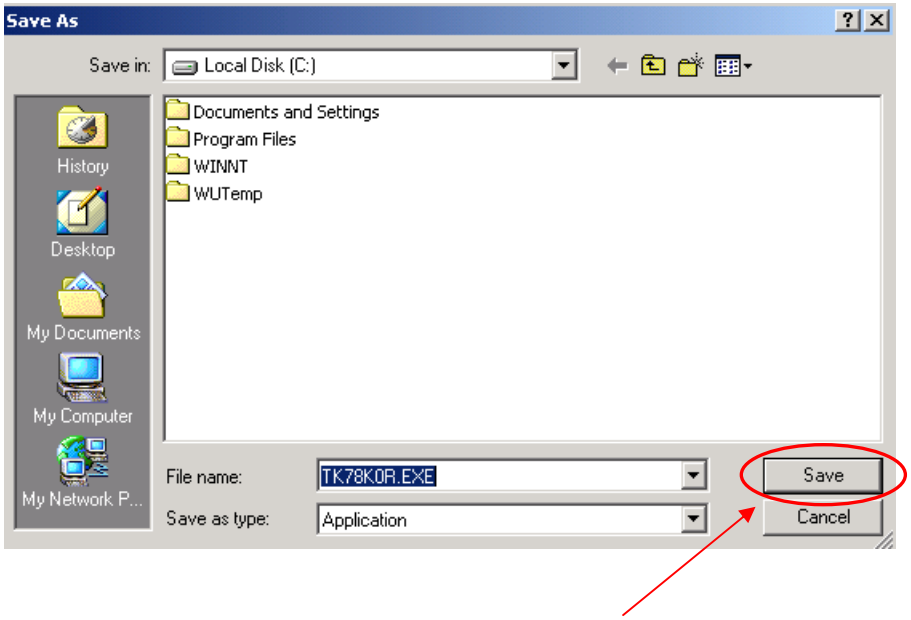
② Press the Sample Program button to start the WWW browser.



③ Click the "TK-78K0R/KG3 Sample Programs" link , the following download confimation window appears.



④ Click the Save butten.

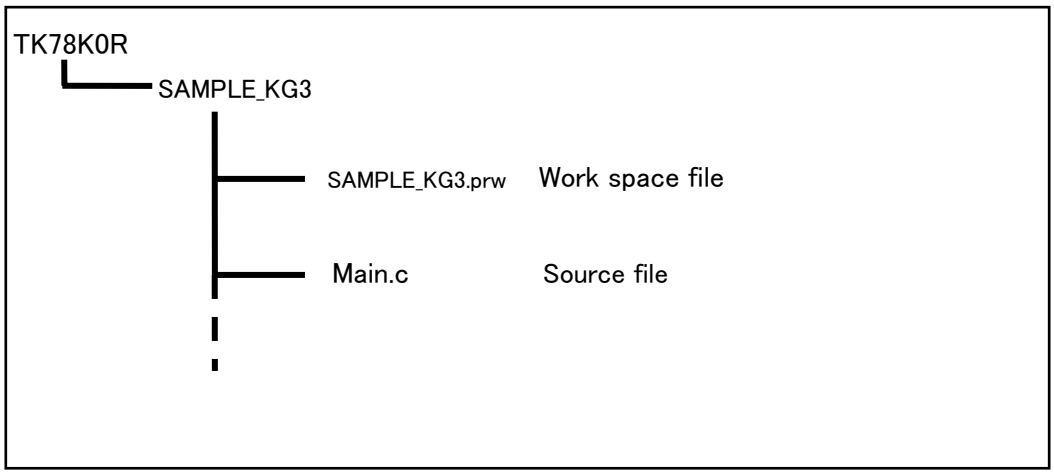


⑤ After specifying the download destination folder, click the Save button.

⑥ The self-extraction sample program set (TK78K0R.exe) is copied to the specified folder.
The folder that the "TK78K0R" folder is made when this file is executed, and the sample program is stored under the folder in addition is made.

1.4.2 Overview of Sample Programs

The sample programs consist of following directories.



CHAPTER 2　Experiences

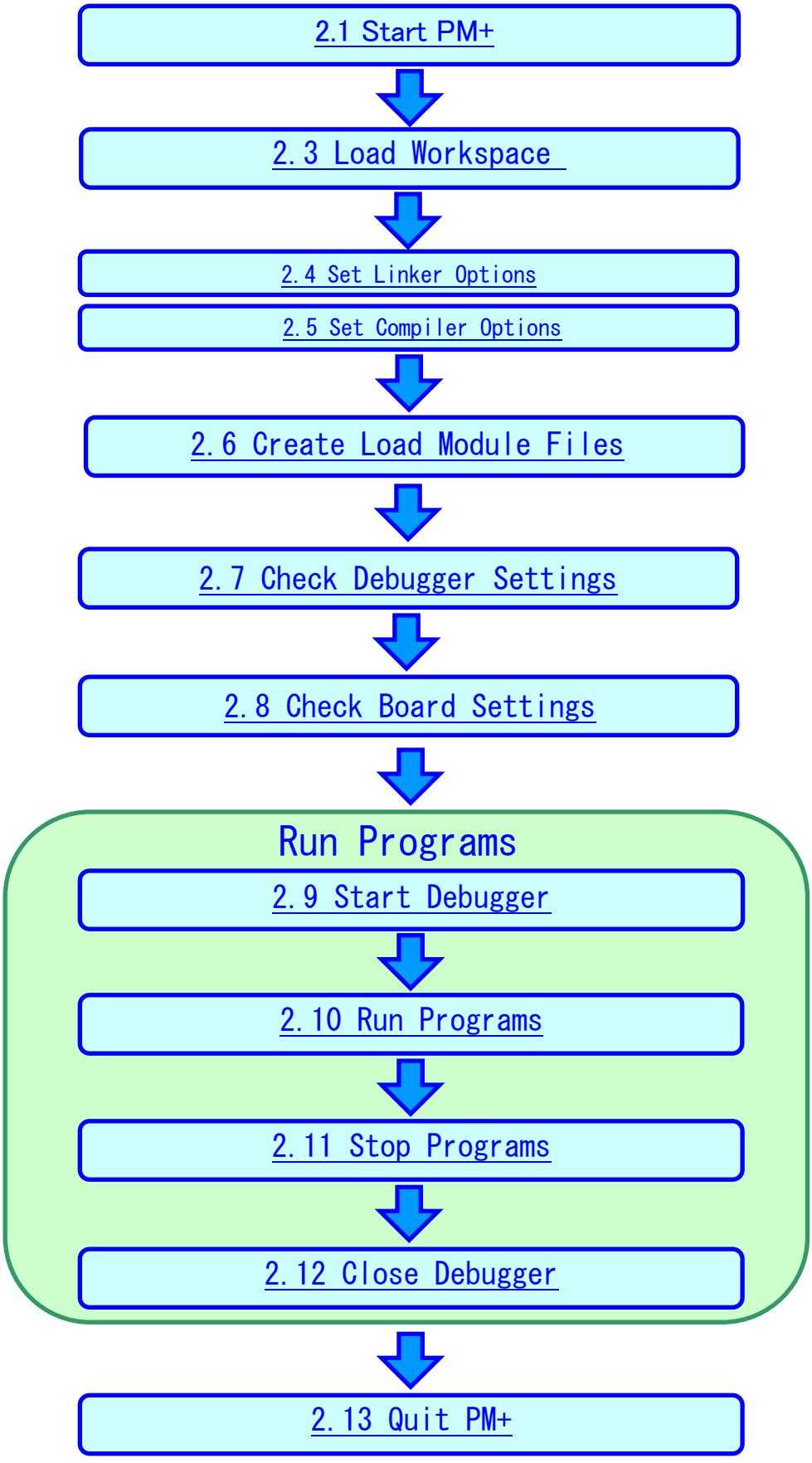
In this chapter, you will experience how to use the development tools with using the sample programs.
The development tools are :

- Integrated Development Environment (IDE), PM+
- Integrated Debugger, ID78K0R-QB

You will use the programs that you prepared in "1.4 Sample Programs", as the sample programs for TK-78K0R/KG3.

You will be able to understand how to use the development tools and the concept of project files which you need for producing application programs.

The overall steps are as follows:



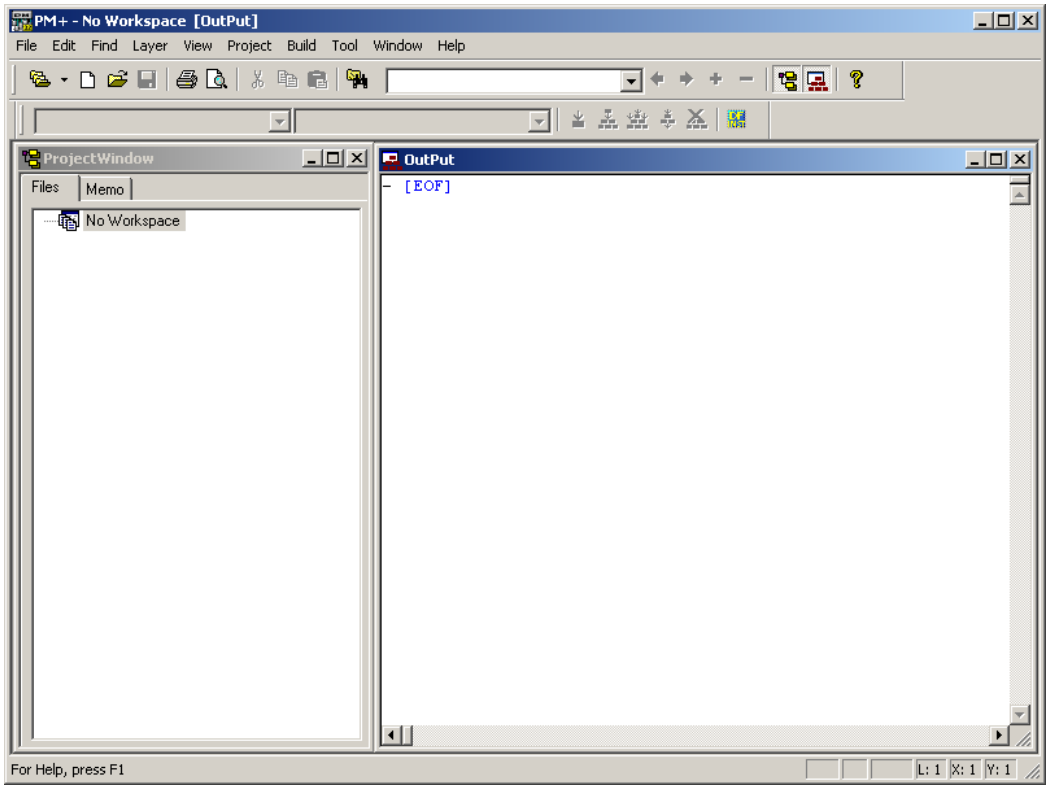
2.1 Start PM+

Let's start using the development tools.

First, start the PM+

Select "Windows Start Menu" -> "Program" -> "NEC Electronics Tools" -> "PM+ V6.30".

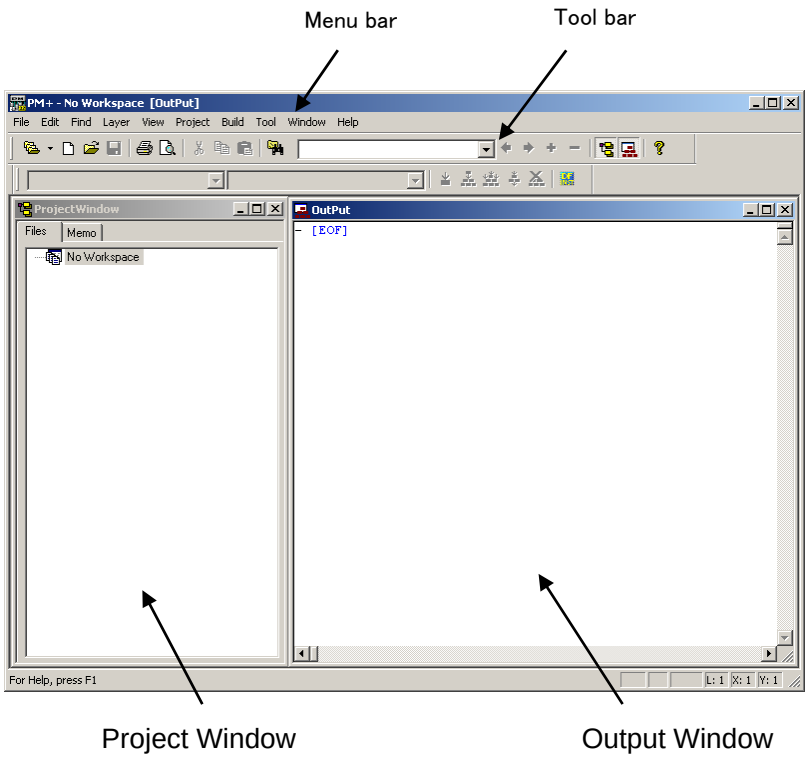
PM+ starts up



2.2 What is PM+

In PM+, application programs and environment setting are handled as a single project, and series of actions such as program creation using the editor, source management, build, and debugging are managed.

Also, one of more project files is managed together as a workspace.



Project window	A window in which project names, source files, and include file are displayed using a tree structure.
Output window	A window in which the build execution status is displayed.

For details regarding menu bars and tool bars, refer to "Help" menu in PM+.
"Help" on menu bar , then "PM+ Help"

What is a project?

A project is the unit that is managed by PM+. A project refers to an application system and environment development based on PM+.

PM+ saves project information in a "project file".

What is a project file?

A project file contains project information that includes the source files, device name, tool options for compiling, editor, and debugger information.

The file name format is "xxxxx.prj".

Project files are created in the directory you specifies when you create a new workspace.

What is a project group?

A project group is a group comprised of a number of projects in an application system.

The target device of each project must be the same within a project group.

What is a workspace?

A workspace is the unit used to manage all the projects and project group required for one application system.

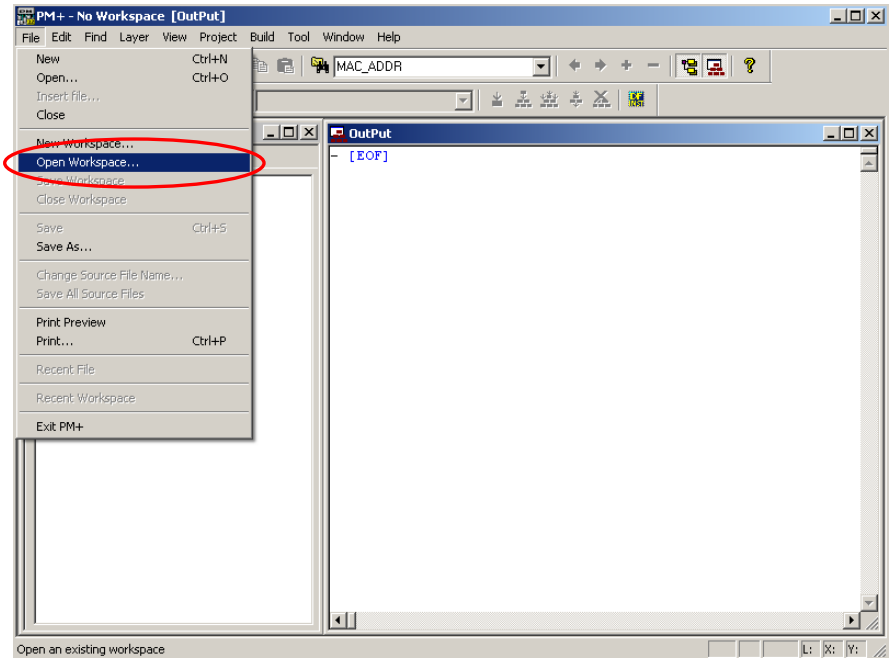
A workspace file contains one or more project files.

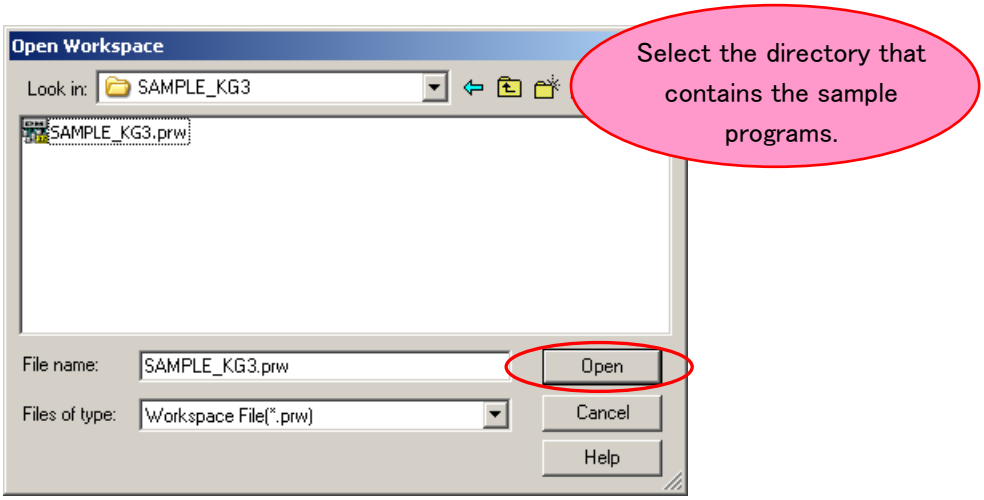
The file name format is "xxxxx.prw".

2.3 Load Workspace (project)

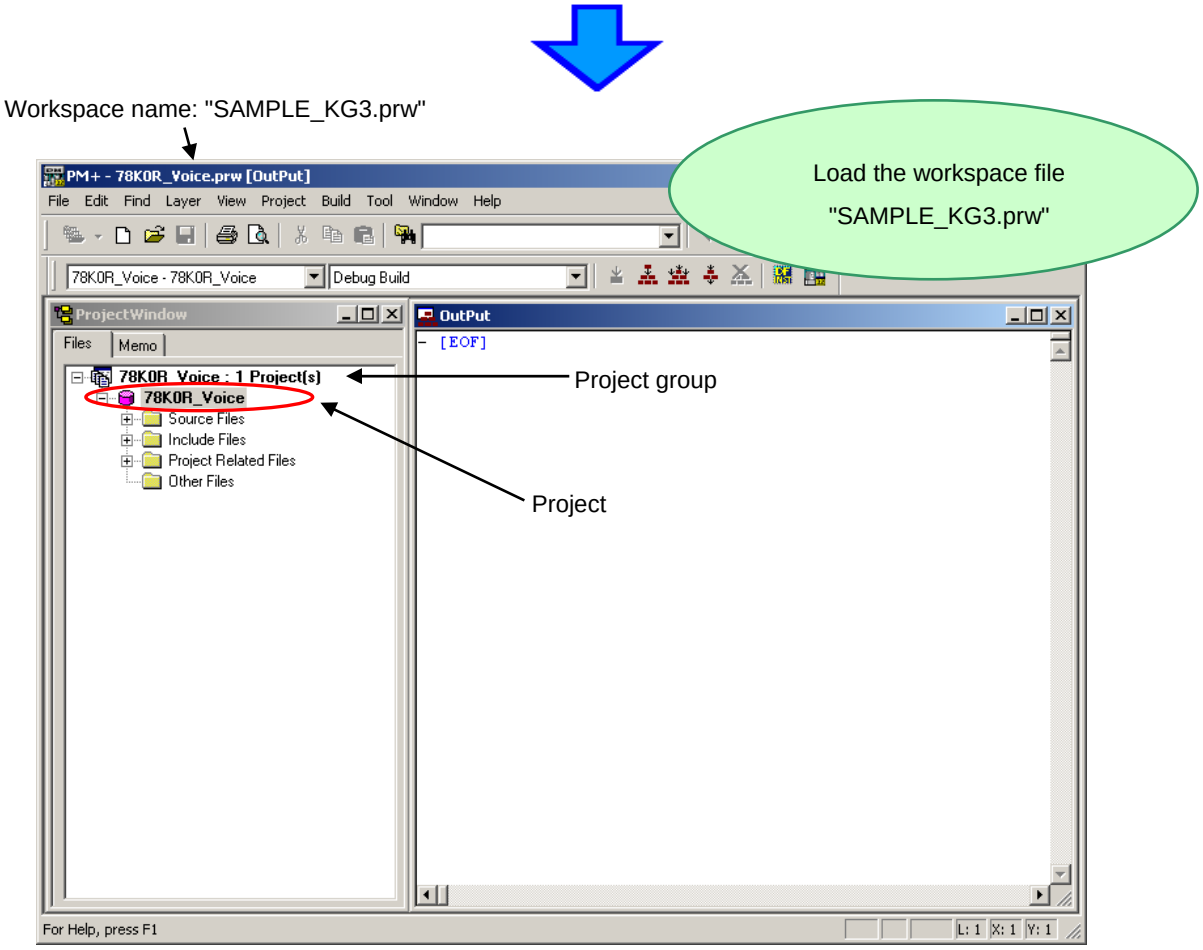
In this section, you will use the workspace that you created in "1.4 Sample Programs"
For creating a new workspace, refer to "Chapter 5 Other Information".
The workspace has information about the build environment for the sample programs.

Select "File" on menu bar and "Open Workspace…".
Then, select "SAMPLE_KG3.prw" under the directory "¥TK78K0R¥SAMPLE_KG3".





Select "SAMPLE_KG3.prw", then click Open .



The workspace file "SAMPLE_KG3.prw" contains one project called "SAMPLE_KG3". You will use this project "SAMPLE_KG3".

CAUTION:

Please ignore when you get a prompt saying "files could not be found". This may occurred when the installation directory is not a default.

2.4 Set Linker Options

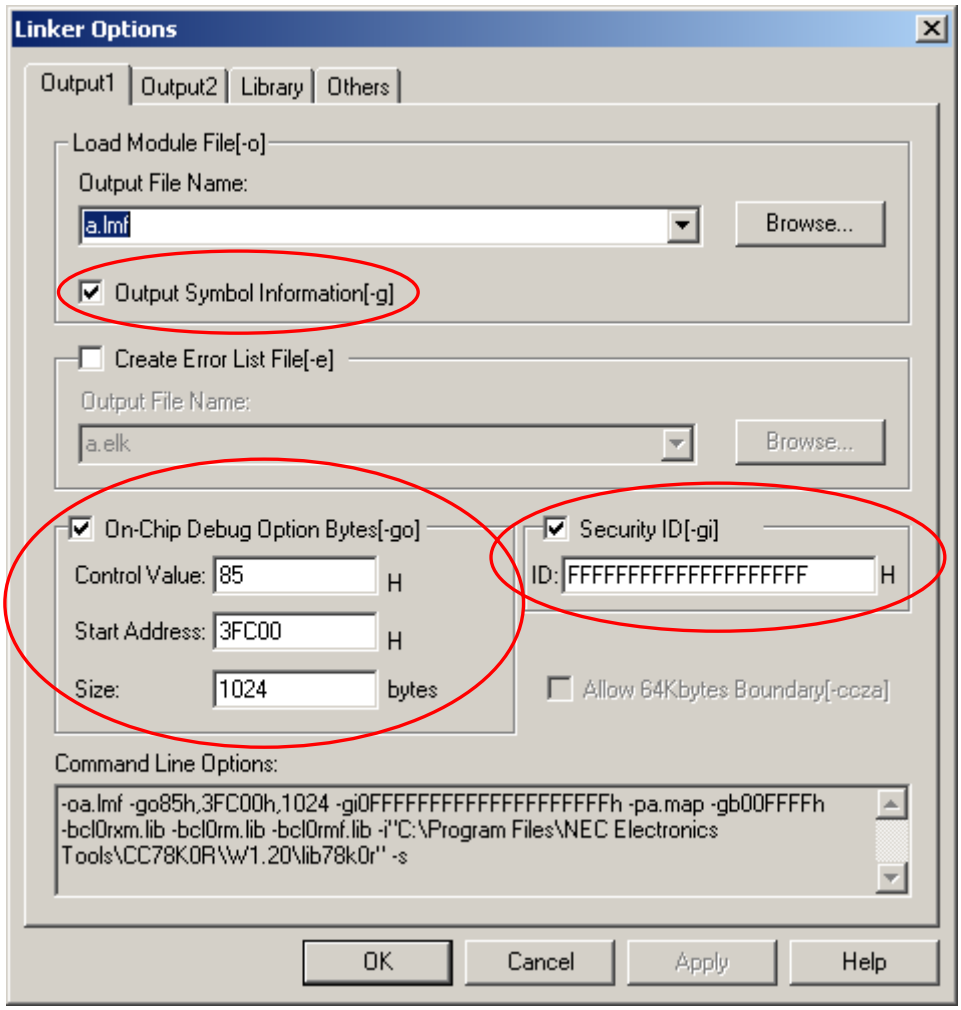
The linker options have been set by the project file. However, some option settings will be covered in this section because the linker option settings are important for debugging. Following three settings are covered specifically.

- Outputs from debugging
- On-chip debug (Desable/Enable, security ID)
- Watchdog timer

Select "Tools" on menu bar, then "Linker options....".

2.4.1 "Output1" Tab

Select "Output1" tab on "Linker Options" window, and see following settings.



- Load Module File settings

  Check "Output Symbol Information".

  This enables to do source level debugging (setting break points, monitoring variables in watch window, etc).

  Also, you can specify the load module file name.

- On-Chip Debug Option Byte

  Check "On-Chip Debug Option Byte". Enter "85" in "Control value". This setting enables the on-chip debugging function of the microcontroller.

  * For details of "Control value", refer to the user's manual of 78K0R/KG3 (U17847E).

  See "Start address" is set to "3FC00", and "Size" is set to "1024".

  These settings reserve the memory address area for the monitor program (the flash memory area that the debugger uses for on-chip debugging).

  In this case, the "Control value" is allocated to the address of C3H in flash memory, and FFH is set to the next address. Because of this, the following areas could not be set the segments.

  <Address area that reserved by on-chip debugging>

  - 2H, 3H
  - CEH-D7H
  - From the address set in "Start address" to the byte set in "Size"

- Security ID

  Check "Security ID", and enter the security ID which is a unique ID code (10 bytes) to authenticate when the debugger is launched.

  The security ID is stored in the flash memory (C4H-CDH), and checked if it is the same as the code entered in Linker options dialog when the debugger is launched.

  The debugger will not be launched when the security ID is unmatched. By using this function, you can secure the programs from leaks.

  If you do not need to set the security, it is recommended to set the security ID "FFFFFFFFFFFFFFFFFFFF" as this is the initial code.

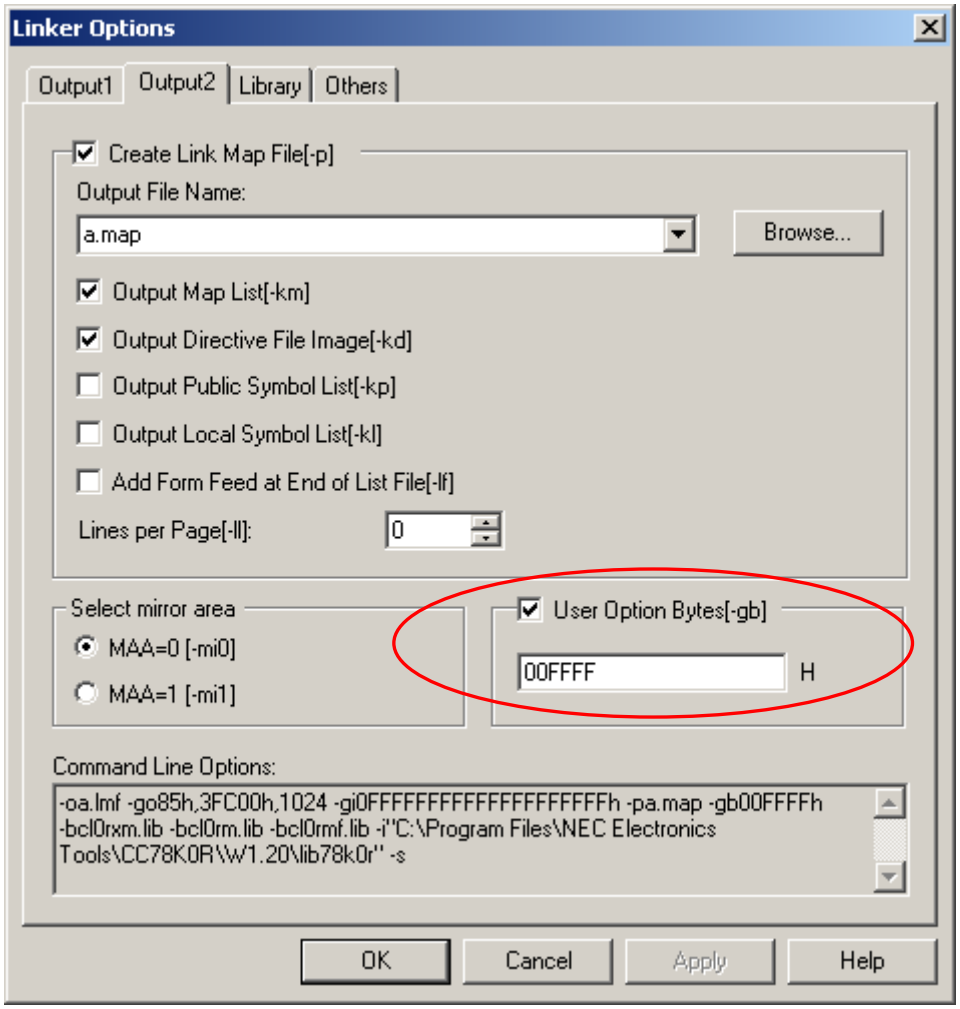
If you forget about the security ID (stored in the address of C4H-CDH) or if you set wrong on-chip debug option byte, you will not be able to use the debugger (ID78K0R-QB).

To solve this, you can use the "78K0R Starter Kit Setting" for changing the setting to initialize the flash memory at the time debugger launched.

For details, refer to "5.3.8 Erase microcontroller built-in flash memory".

2.4.2 "Output2" Tab

Select "Output2" tab on "Linker Options" window, and see following settings.



- User Option Byte

  Check "User Option Byte", and then enter "00FFFF". Here, you can do the setting of watchdog timer, low-voltage detector, and system reserved memory area. The 3 bytes you entered are stored at C0H-C2H on flash memory:

  - C0H: setting for watchdog timer
  - C1H: setting for low-voltage detector
  - C2H: setting for system reserved memory area (must be set as FFH)

  This time, you disabled the watchdog timer and the default start function of low-voltage detector. For details, refer to the user's manual, 78K0R/KG3 (U17847E).

2.5 Set Compiler Options

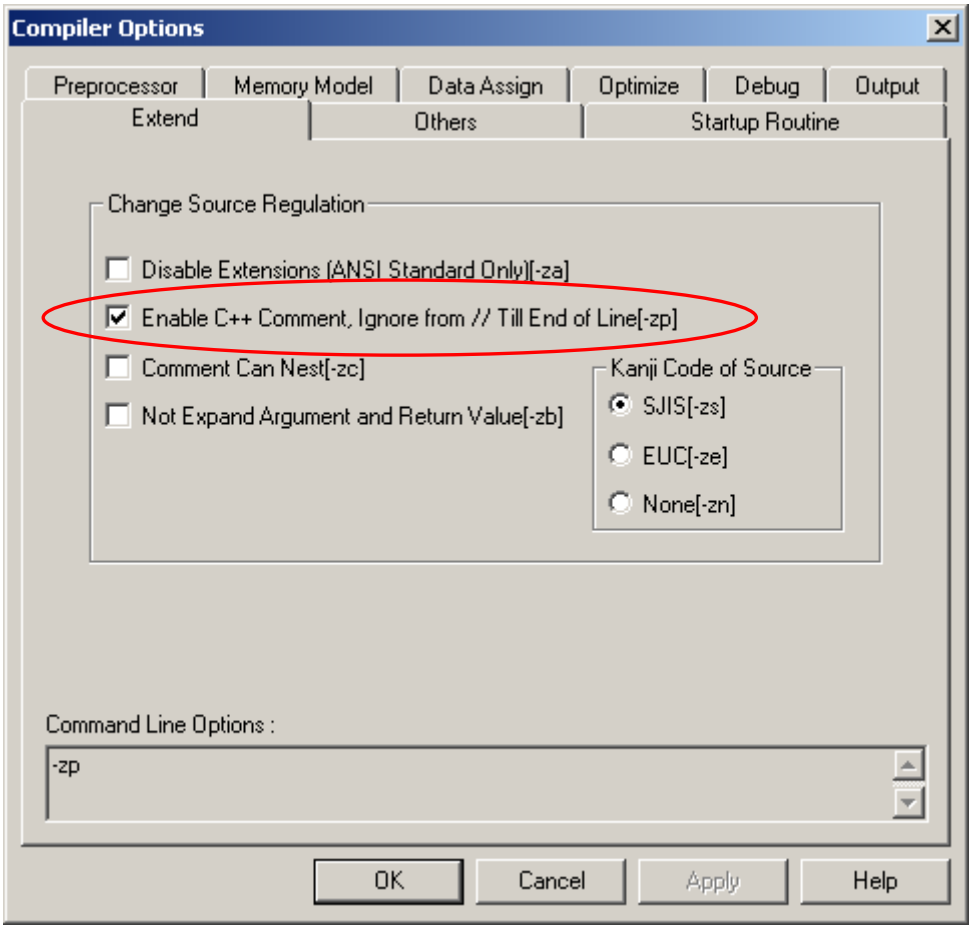
The compiler options have been set by project file. However, because some compiler options are useful, following two settings are covered specifically in this section.

- Enable C++ comments
- Use multiplier and divider

Select "Tools" on menu bar, then "Compiler options".

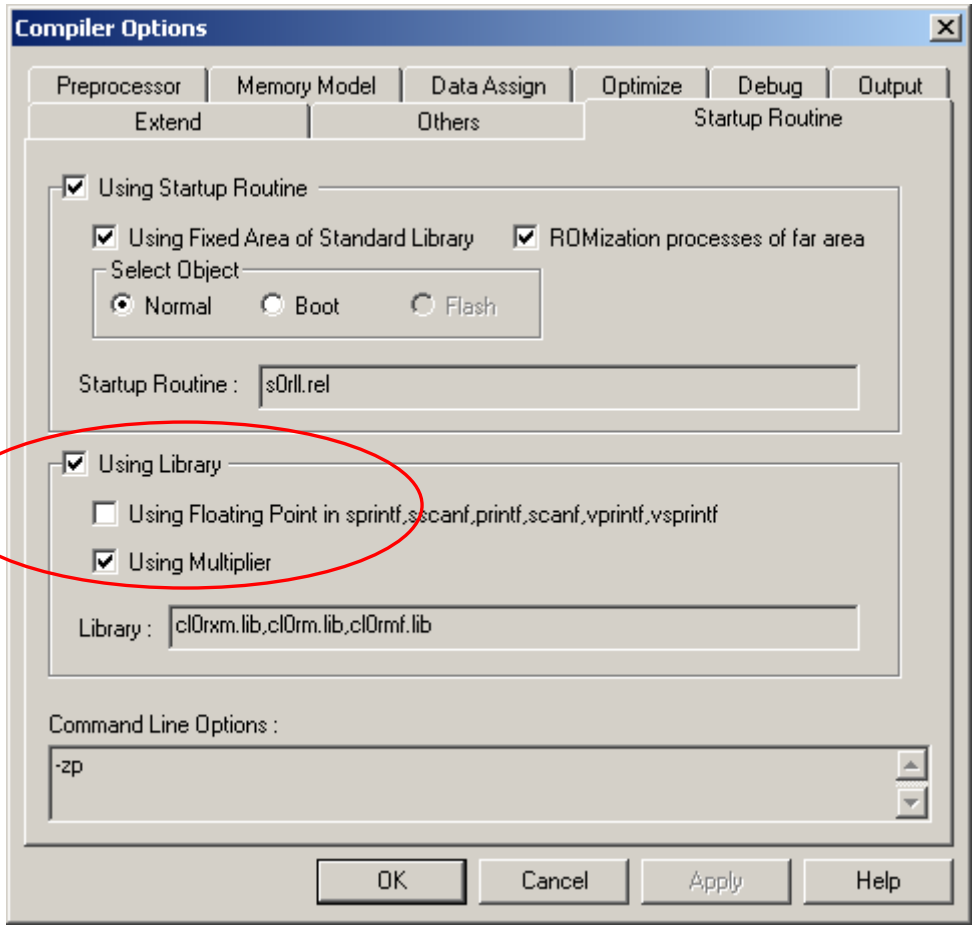
2.5.1 "Extend" Tab

Select "Extend" tab, and check "Enable C++ Comment".
This setting allow you to use the C++ comment using "//".
It is useful feature when developing code.



2.5.2 "Startup Routine" Tab

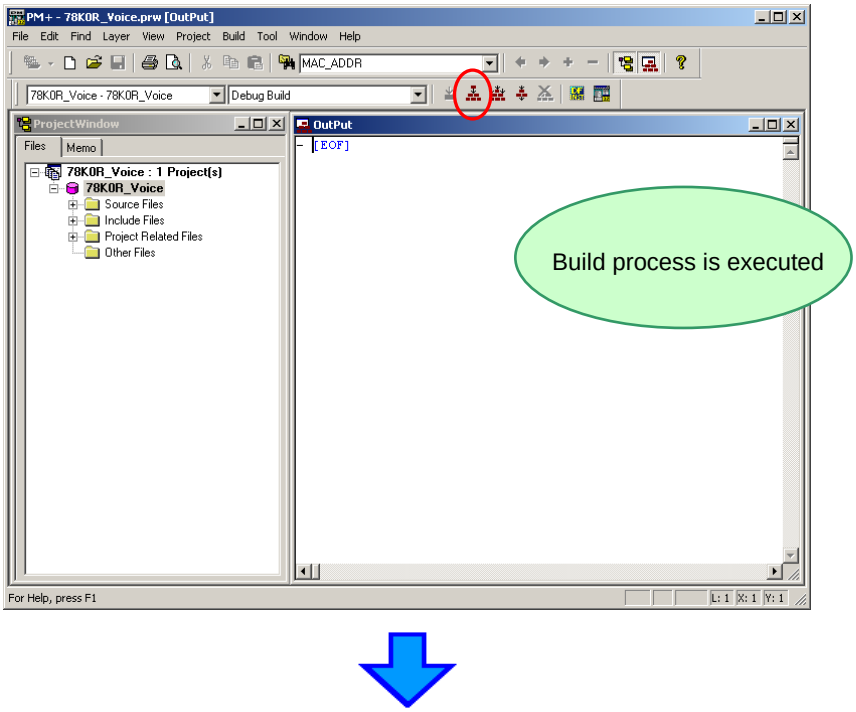
Select "Startup Routine" tab, and check "Using Library" and "Using Multiplier". The 78K0R/KG3 has feature of multiplier to increase those calculation speed.

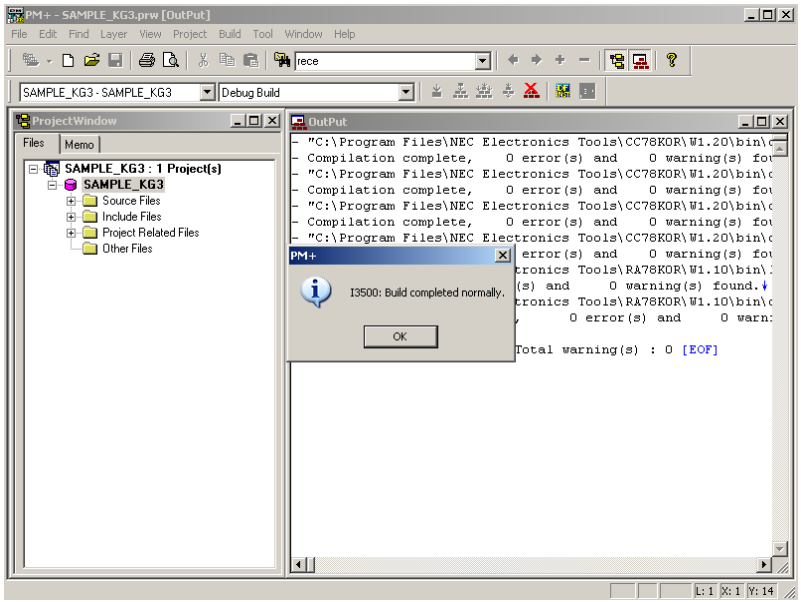


2.6 Create Load Module Files

After developing the source code, you have to create load module files by compiling, assembling, and linking. This process is called build.

Click the build button , or select "Build" on menu bar, then "Build".





Build has been completed successfully.

What is build?

Build is a function that creates an executable file from source files in a project.

PM+ automatically performs compiling, assembling, linking, and other processing actions.

To reduce the time for the build, PM+ detects and compiles/assembles only the files that have been updated from the previous build process.

What is rebuild?

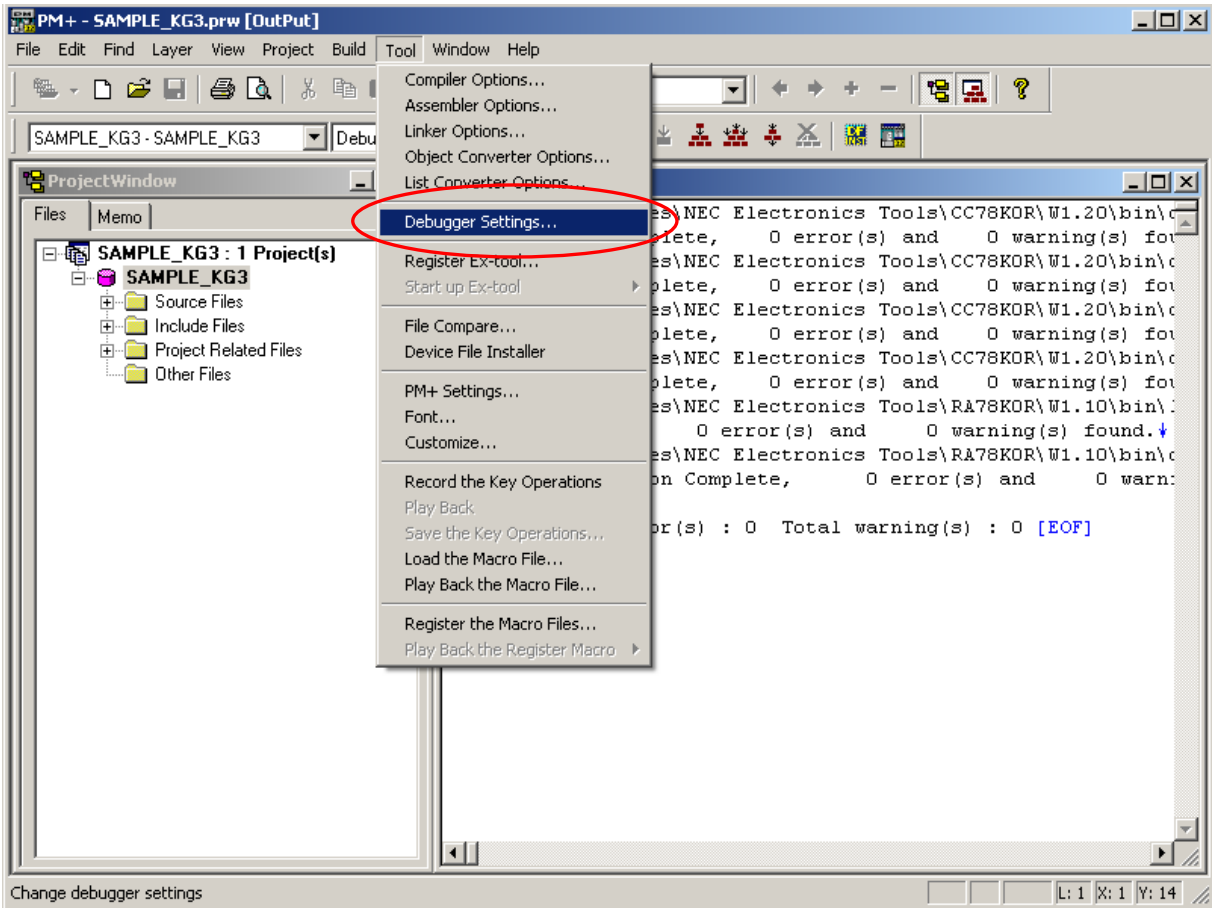
Build compiles and assembles only the source files that have been updated from the previous time, whereas rebuild compiles and assembles all the source files.

When setting, such as compiler options, have been changed, you must rebuild instead of build.

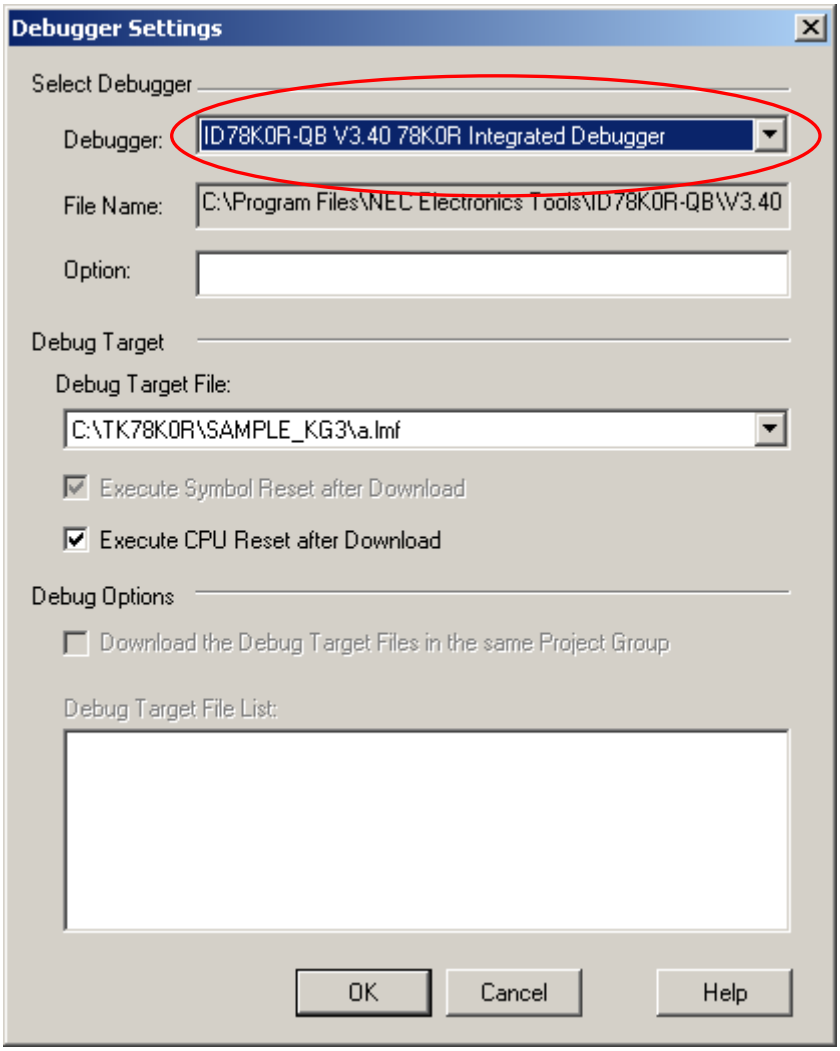
2.7 Check Debugger Settings

After the build, you should configure the debugger settings.
The debugger settings have been set by the project file as well. However, because those settings are important for debugging, some settings are covered in this section.

Select "Tools" on menu bar, then "Debugger Setting...".



Check if "ID78K0R-QB V3.40 78K0R Integrated Debugger" is selected on "Debugger".



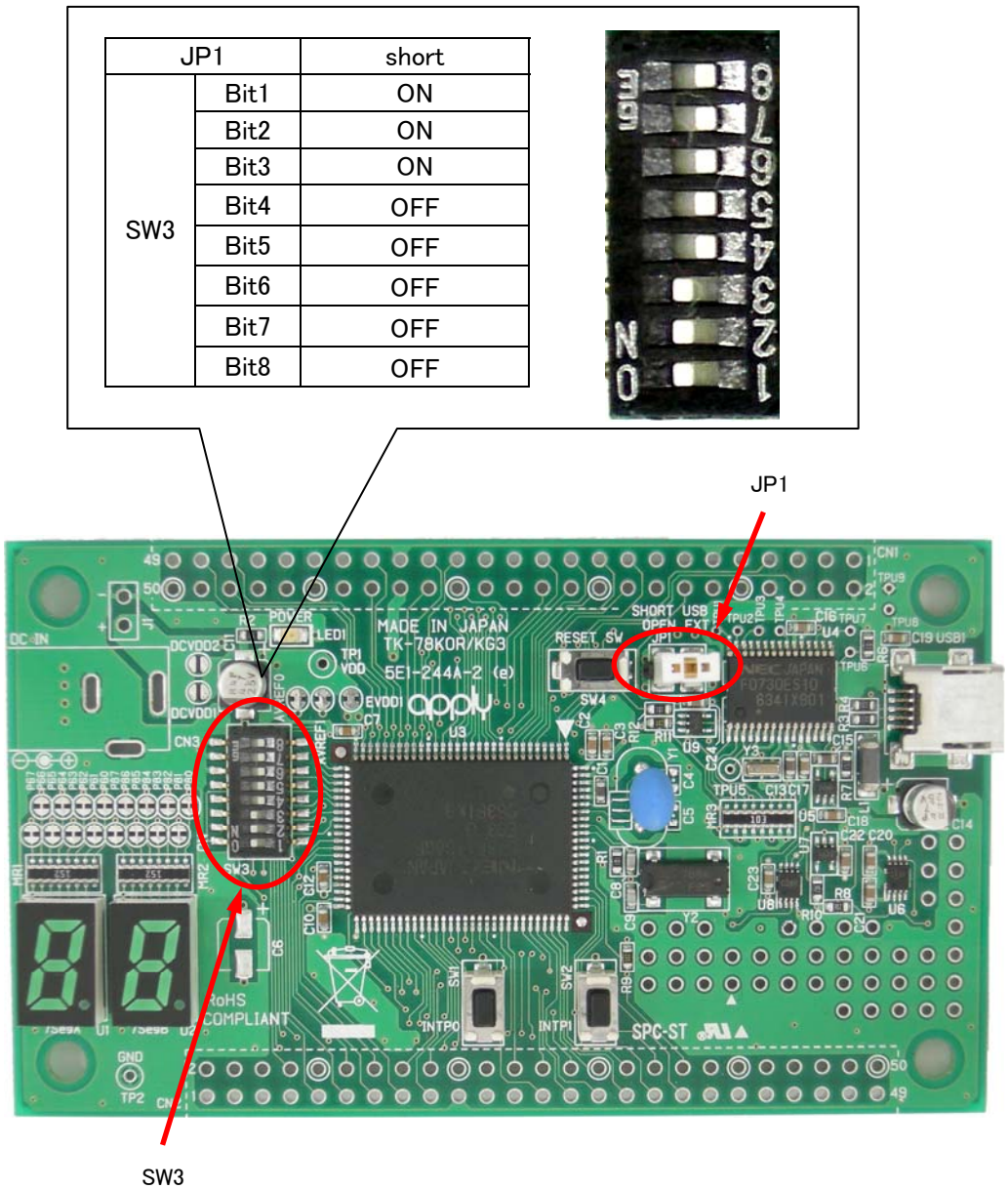
If you cannot select "ID78K0R-QB V3.40 78K0R Integrated Debugger", select "Project" on menu bar, "Project settings" -> "Tool version settings" -> "Detailsetting" -> then select "ID78K0R-QB".

2.8 Check Board Settings

Before connecting the PC and the TK-78K0R/KG3 with USB, you should check the setting of SW3 and JP1 on the board.

Set the SW3 and JP1 of the TK-78K0R/KG3 as follows.

JP1		short
SW3	Bit1	ON
	Bit2	ON
	Bit3	ON
	Bit4	OFF
	Bit5	OFF
	Bit6	OFF
	Bit7	OFF
	Bit8	OFF



After the switch settings are completed, connect the PC to USB1 on TK-78K0R/KG3 with USB cable.

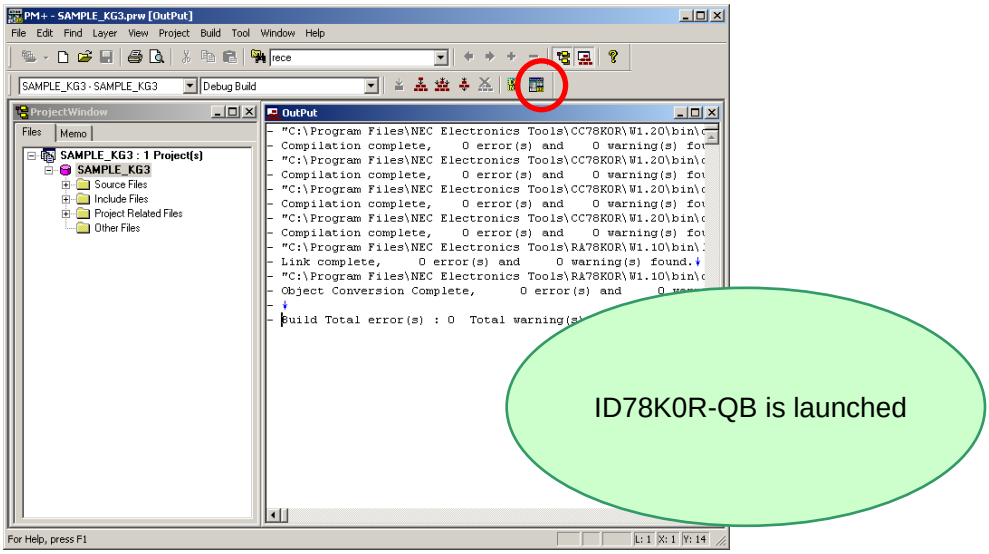
If the "Found New Hardware Wizard" is started, install USB driver with referring "1.3 Installation of USB Driver".

2.9 Start Debugger (ID78K0R-QB)

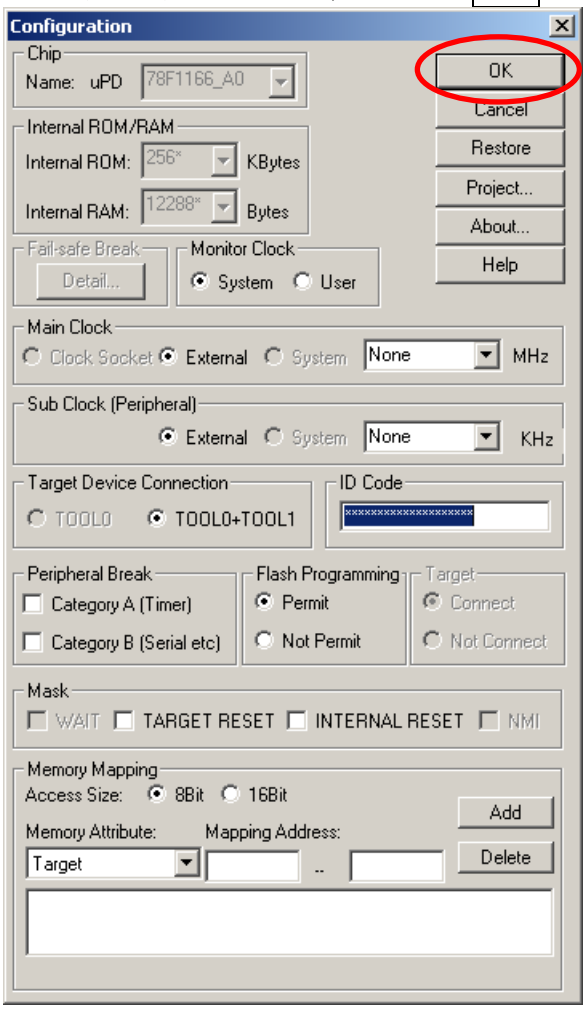
Click the debug button , or select "Build" on menu bar, then "Debug".

If you do not see the debug button, go to "2.7 Check Debugger Settings" for changing the settings.

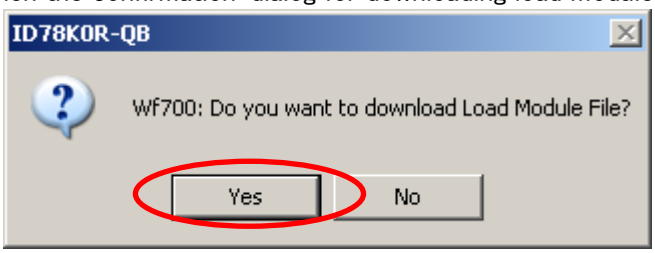
The steps to start the debugger will be explained below.

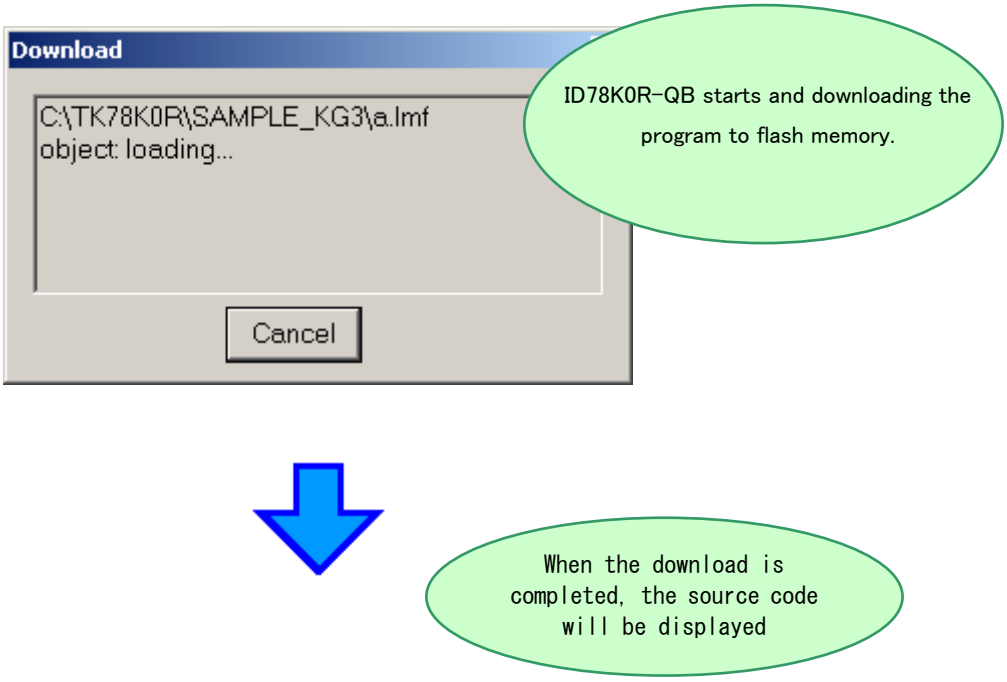


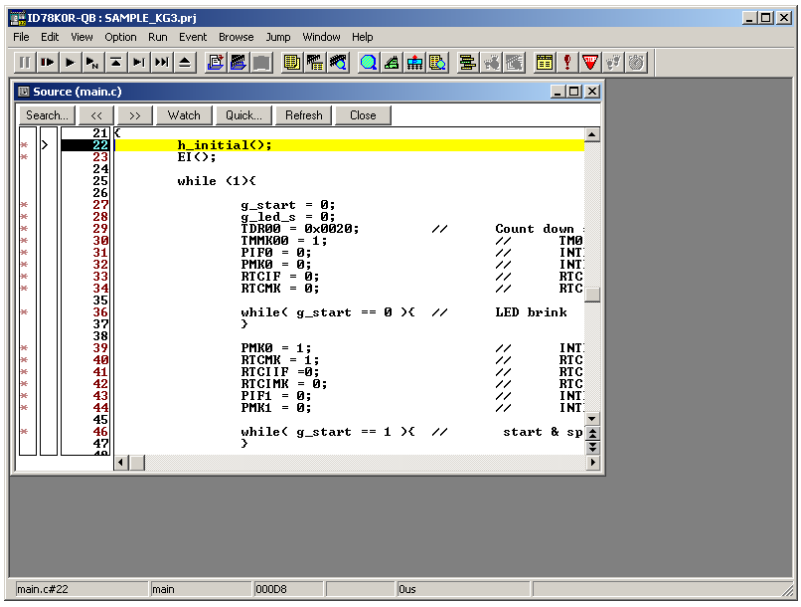
"Configuration" dialog is opened.
Enter "FFFFFFFFFFFFFFFFFFFF" (F x 20) in "ID Code", then click OK .



Click Yes when the confirmation dialog for downloading load module file is opened.







NOTE:

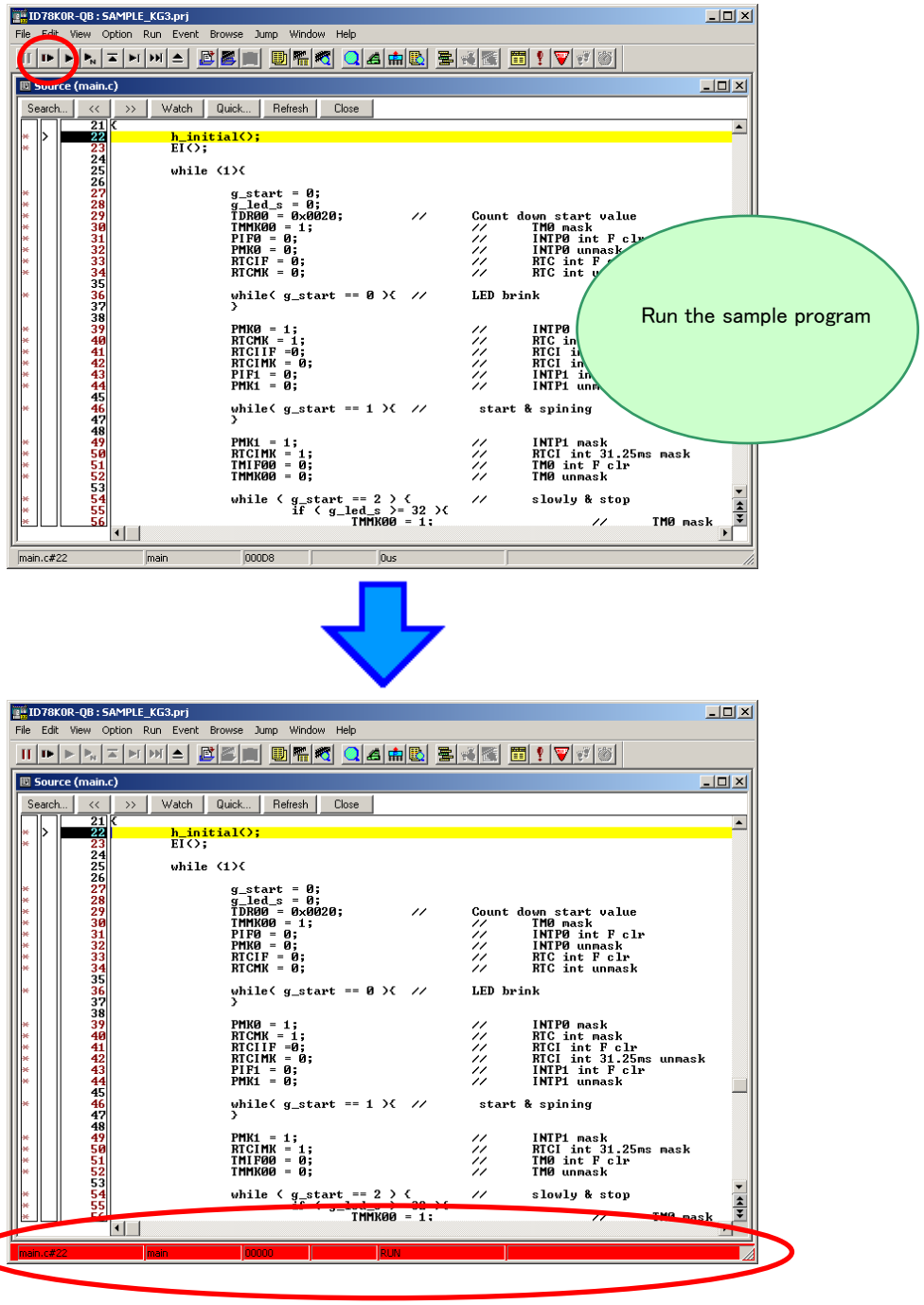
Completion of the download does not mean running the programs. Therefore, even though you press switch on the board, it does not make anything happened. To run the sample program demonstration, see "2.10 Run Programs".

2.10 Run Programs

Now, you are ready to run the program.

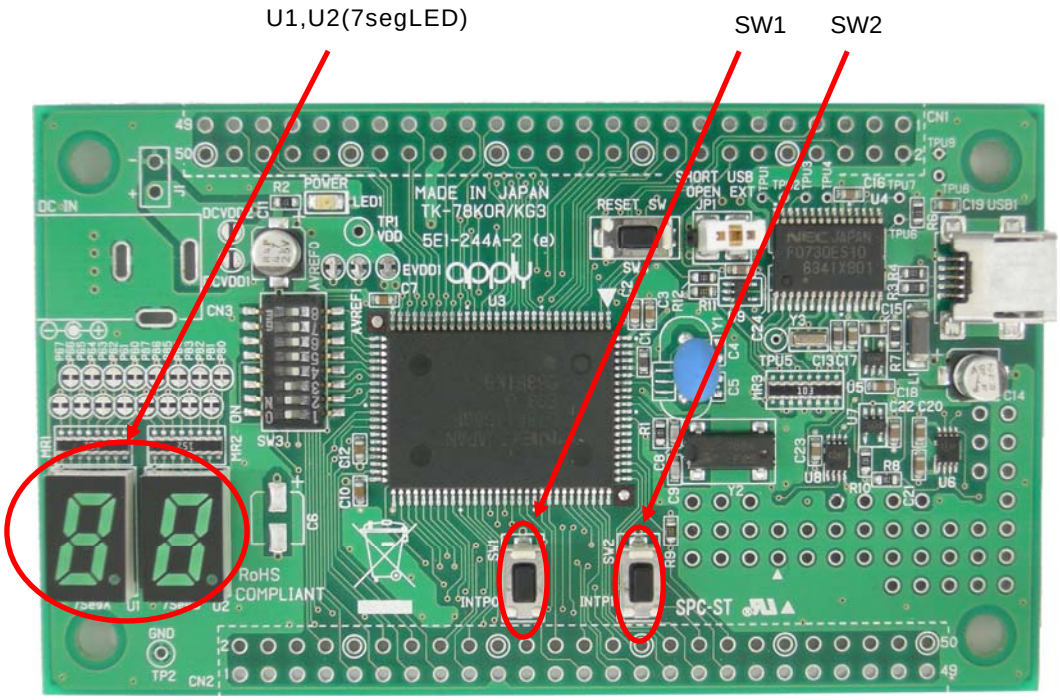
Click the restart button , or select "Run" on menu bar, then "Restart".

The sample program runs.



When programs are running, the status bar will be red.

If you push SW1, the 7-seg LED segments rotates.
You push the SW2. and confirm the display of number after a while.

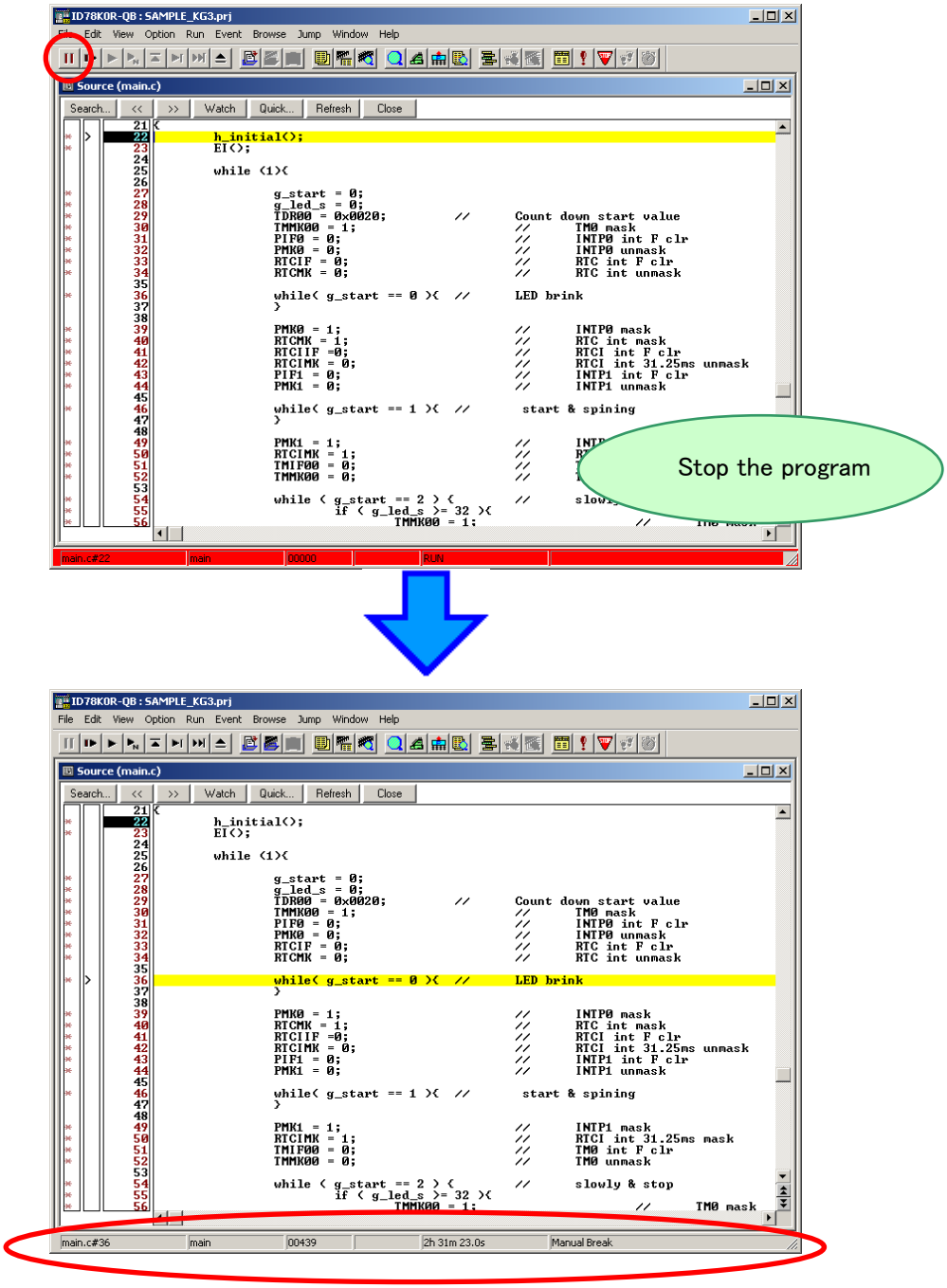


You could confirm the sample program is working.

2.11 Stop Programs

Now, you are going to stop the program.

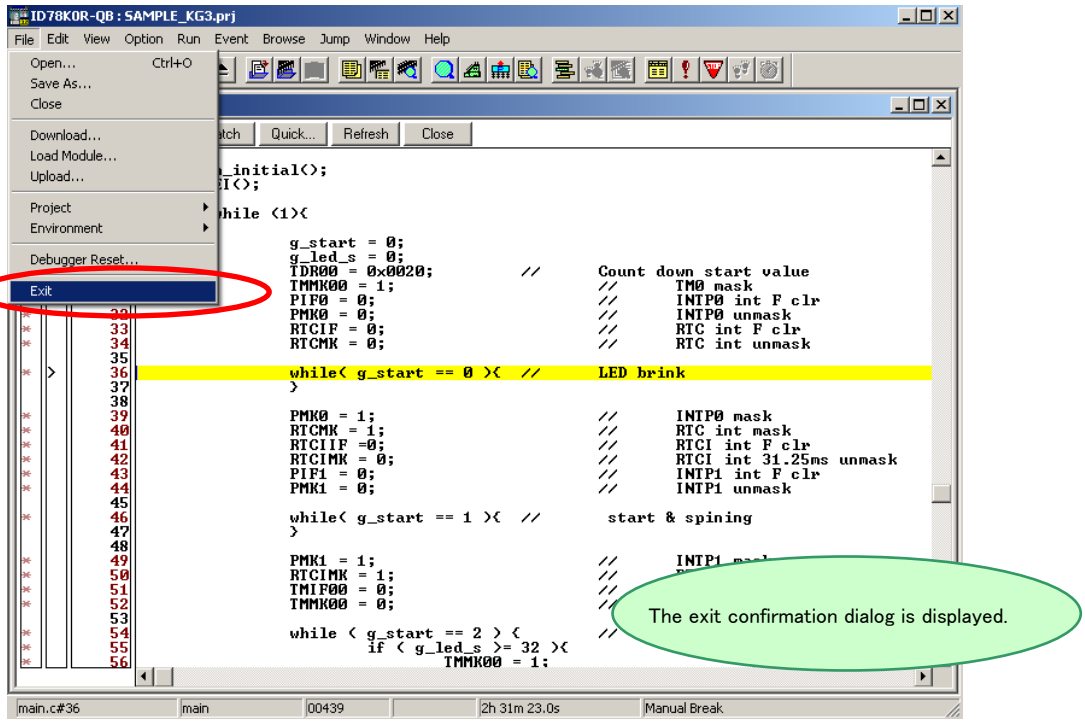
Click the stop button , or select "Run" on menu bar, then "Stop".

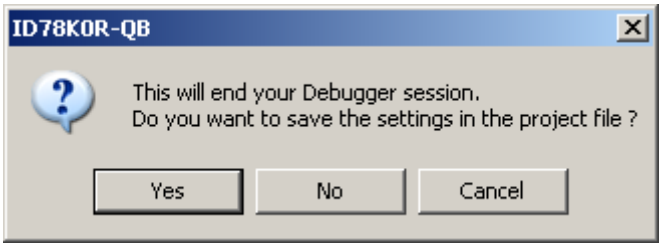


When the program stops, the status bar changes back to the original color.

2.12 Close Debugger (ID78K0R-QB)

Select "File" on menu bar, then "Exit".



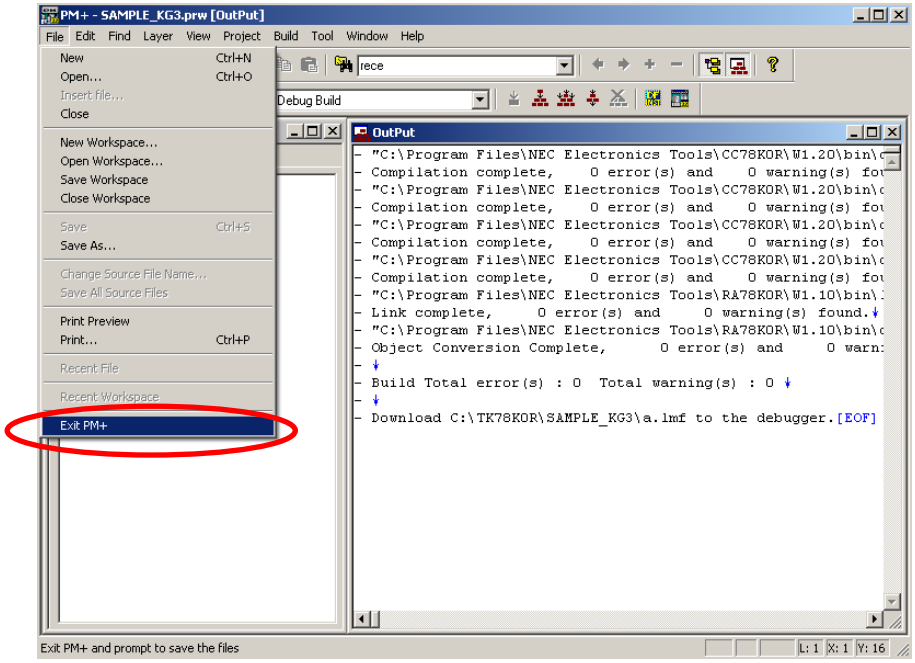


If you click [Yes] , it saves the settings in the project file, and then closes the ID78K0R-QB. It is recommended to save the settings as it saves the window you used, window size, layout, etc.

If you click [No] , it does not save the settings and closes the ID78K0R-QB.

2.13 Quit PM+

Select "File" on menu bar, then "Exit PM+".



PM+ is closed.

The experiences section ends now.
You can find more information how to use the development tool and information about other useful features in "Chapter 5 Other Information".

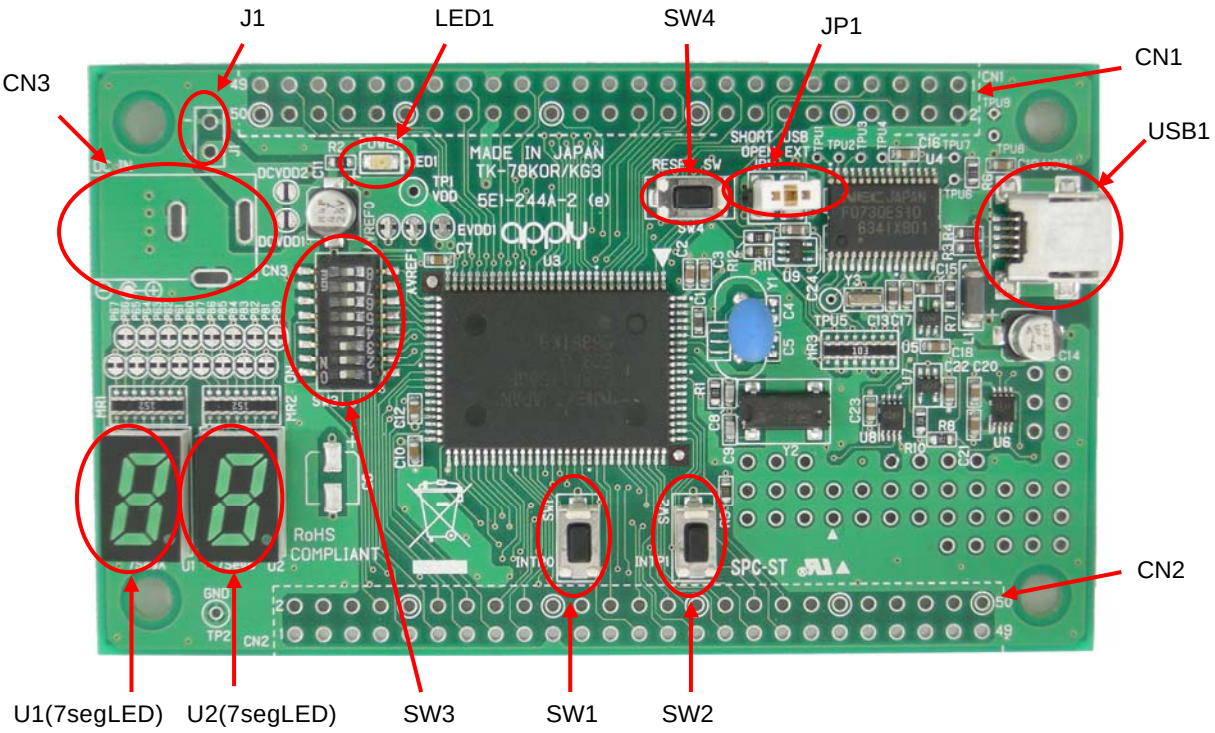
CHAPTER 3　Hardware Specifications

In this chapter, the hardware of TK-78K0R/KG3 will be explained.

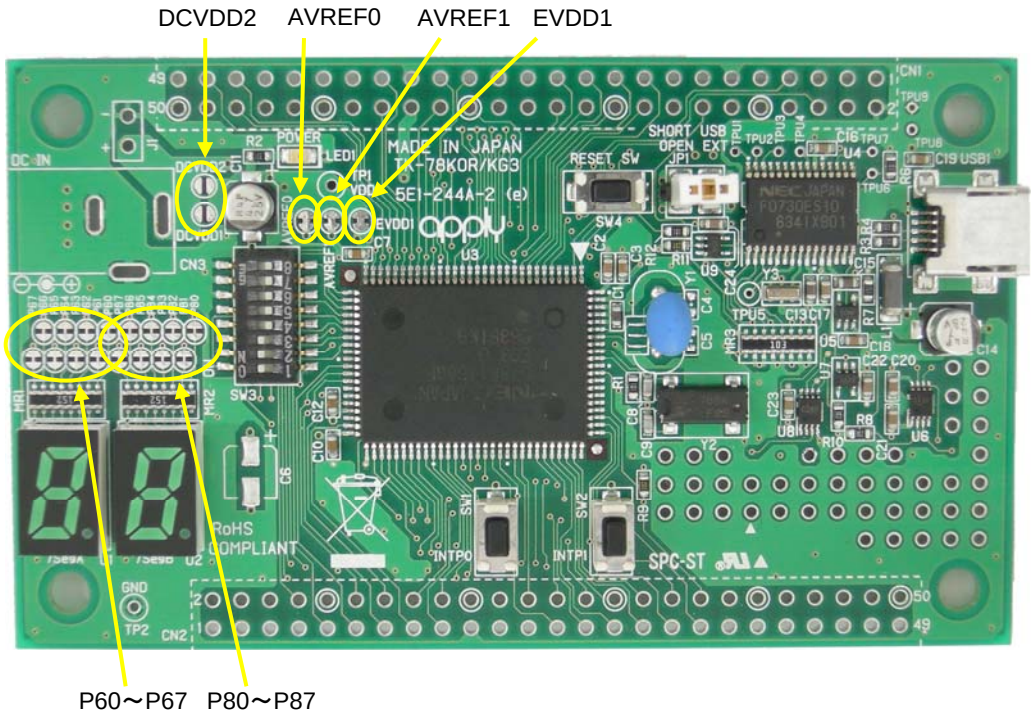
Microcontroller	μPD78F1166GF　*78K0R/KG3
Clock	External main system clock: 20MHz Subsystem clock: 32.768KHz Internal high-speed oscillation clock: 8MHz Internal low-speed oscillation clock: 240 kHz
Interface	USB (USB1) Expansion connector 50Pin socket x2 (only pad)
Power supply voltage	5V
Input/output for operation check use	・7-Seg LED (U1,U2) ・Push switch (SW1,SW2) ・Reset switch (SW4)
Other parts	・Operating mode setting switch (SW3) ・Power supply selecting jumper pin (JP1) ・Power LED (LED1) ・solder short pad

* The name with bracket is the name printed on the board.

3.1 Layout of hardware functions



3.2 Layout of solder-short pad



3.3 Hardware Functions

3.3.1 SW1

SW1 is the push switch connected to "P120/INTP0/EXLVI" pin in CPU. When the switch is pushed down, it sends the signal of "Low". When it is released, it becomes "Open". Therefore, when you need to use this, you need to set the microcontroller built-in pull-up resistor option registers (PU12) to ON. For details about settings of microcontroller built-in pull-up resistor option registers, refer to 78K0R/KG3 User's Manual (U17894E).

3.3.2 SW2

SW2 is the push switch connected to "P46/INTP1/TI05/TO05" pin in microcontroller. When the switch is pushed down, it sends the signal of "Low". When it is released, it becomes "Open". Therefore, when you need to use this, you need to set the microcontroller built-in pull-up resistor option registers (PU4) to ON. For details about settings of microcontroller built-in pull-up resistor option registers, refer to 78K0R/KG3 User's Manual (U17894E).

3.3.3 SW3

The bit 1-5 on SW3 are for mode settings, and bit 6-8 are DIP switches connected to P50-P52 pins in microcontroller.

● For the use of ID78K0R-QB, use following settings.

SW3

Bit 1	ON/OFF ※1
Bit 2	ON
Bit 3	ON
Bit 4	OFF
Bit 5	OFF

***1 ON: The microcontroller stays being reset until ID78K0R-QB is started.**
OFF: The microcontroller runs the programs stored in the flash memory as soon as it gets power supply.

● To run the programs stored in built-in flash memory without using ID78K0R-QB, use following settings and re-supply power.

SW3

Bit 1	OFF
Bit 2	OFF
Bit 3	OFF
Bit 4	OFF/ON ※2
Bit 5	OFF/ON ※2

***2 ON: Serial connection can be made with using UART3 (P13(TXD3), P14(RXD3)) on microcontroller.**

● The bit 6-8 are connected to P50-P57 pins in microcontroller. ON means "Low" and OFF means "Open". When you need to use this, you need to set the microcontroller built-in pull-up resistor option registers (PU5) to ON. For details about settings of microcontroller built-in pull-up resistor option registers, refer to 78K0R/KG3 User's Manual (U17894E).

SW3

Bit 6	P50
Bit 7	P51
Bit 8	P52

3.3.4 SW4 (RESET SW)

This is the reset switch. You can reset the microcontroller by pressing this switch.

3.3.5 LED1 (POWER)

This is the POWER LED. It is lighted when it gets power supply.

3.3.6 JP1

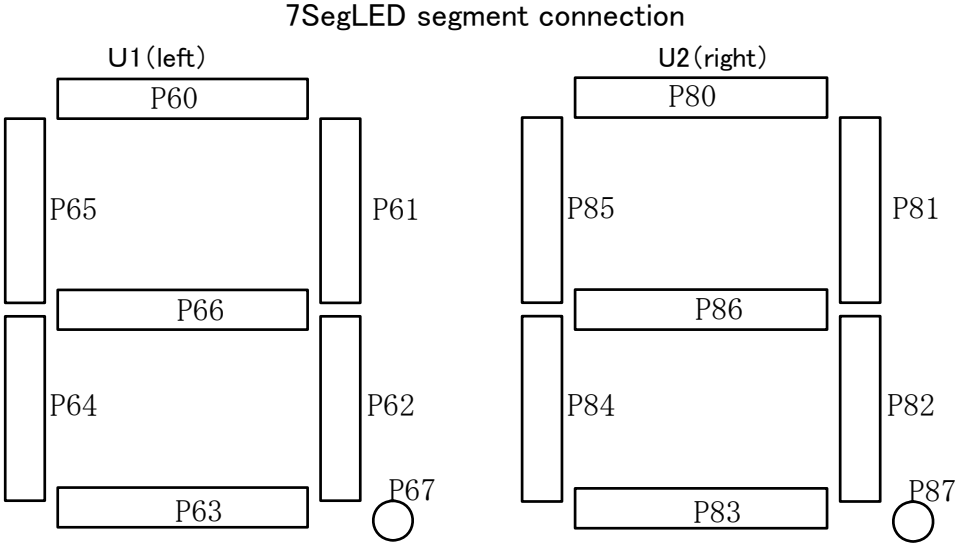
JP1 is the jumper switch pin to select power supply.

JP1

Short	Use power supply from USB power connected to USB1
Open	Use power supply from other (expect USB)

3.3.7 U1,U2(7seg LED)

U1, U2 are general purpose 7SegLED. LED is lighted when each port outputs "Low".



The figure of 0-9 can be displayed by writing the following values in P6 or P8 register.

Example of displayed figure and set data.

0	0xC0	5	0x92
1	0xF9	6	0x83
2	0xA4	7	0xf8
3	0xB0	8	0x80
4	0x99	9	0x98

3.4 Universal Area

The kit has the universal area. Users can use this to develop custom circuit.

3.5 Solder-short pad label

With using the solder-short pad to cut the circuit, users can customize the circuit.
The solder-short pad looks like the picture below.
To open, use cutter to cut the dent part. To short, put solder on the pad.

Solder-short pad
(Open)

Solder-short pad
(Short)

Solder-short pad name	Before Shipment	Connection	
P60～P67	Short	Short	7-seg LED via 1.5k ohm resister
		Open	You can use port6 for multipurpose I/O.
P80～P87	Short	Short	7-seg LED via 1.5k ohm resister
		Open	You can use port8 for multipurpose I/O.
AVREF0, AVREF1	Short	Short	VDD = AVREF0 = AVREF1
		Open	You can Supply different voltage than VDD.
EVDD1	Short	Short	VDD =EVDD0=EVDD1
		Open	You can Supply different voltage than VDD
DCVDD1,DCVDD2	Open	Short	CN3、J1 is connected to VDD.
		Open	CN3、J1 is separated from VDD.

Note: All ground signal are each connected.
Attention: If you supply VDD(5V) from CN3 or J1, disconnect JP1 jumper pin.

3.6 Operation by external power supply

The power supply of the AC adaptor connected with CN3 is connected only with the power supply terminal of the connector of the board in the surrounding (10, 12, and 16pin of CN1). However, not to tie to USB this board and to operate with the unit, the AC adaptor can be made a power supply by the connection of the AC adaptor of 5V to CN3 and the short-circuit of solder short pad (DCVDD1,2). Moreover, it is also possible to connect the stabilizing supply etc. in the lead line instead of the AC adaptor because CN3 is connected with J1 of a through hall.

- Acceptable jack (CN3) : HEC0470-01-630 by Hosiden Corp(not mounted)
- Acceptable plug : 2.1mm DC jack(center plus)
- Current capacity : 100mA or more

Attention : Please make JP1 Open when operating in an external power supply

CHAPTER 4 Troubleshooting

This chapter describes how to solve troubles you may face.

4.1 If you cannot find USB driver when you connect PC to the kit

Check Point 1

If you use USB hub, do not use it. (USB hub is not supported)

Check Point 2

Check if you installed "Starter Kit USB Driver" in "1.2 Installation of Development Tools". If not, install the driver.

Check Point 3

If above 2 check points are confirmed, disconnect the USB cable from PC and re-connect again. It should show the "Found New Hardware Wizard" wizard. Operate the installation with referring to "1.3 Installation of USB Driver". After the installation, make sure you go through "1.3.3 Completion of USB Driver Installation" to confirm the USB driver installation.

4.2 Error when you start the debugger

There could be several reasons to make errors happen.
The solving processes differ depending on errors. Please check the error message first.
The solving processes for each error are as follows.

4.2.1 "Can not communicate with Emulator..." (F0100 or A0109)

Check Point 1

If you use USB hub, do not use it. (USB hub is not supported)

Check Point 2

Check if the settings of switches on the kit are correct with referring "2.8 Check Board Settings".

Check Point 3

Confirm the USB driver installation with referring to "1.3.3 Completion of USB Driver Installation".

Check Point 4

If above 3 check points are confirmed, close the debugger and disconnect the USB cable from PC. Re-connect USB cable properly to both the PC and the kit, and then re-start the debugger.

4.2.2 "Incorrect ID Code." (Ff603)

This error occurs when the security ID stored on microcontroller built-in flash memory is different from the ID code you entered at the start of debugger.

Security ID entry area at the start of debugger

Check Point 1

Enter correct security ID and click OK on the configuration window.

Check Point 2

If you forgot the security ID, you have to erase the microcontroller built-in flash memory. Before erasing, check if you actually set the security ID with referring to "2.4 Set Linker Options". Also remember the code you set for the security ID.
After this, erase the flash memory with referring to "6.3.8 Erase microcontroller built-in flash memory".

4.2.3 "The on-chip debug function had been disabled in the device." (F0c79)

This error occurs when the value at address C3H (On-chip debug option byte) in microcontroller built-in flash memory is incorrect. You need to erase the flash memory.

Check Point 1

Check if you actually set the correct on-chip debug option byte with referring to "2.4 Set Linker Options". If it is not correct, then set correctly.

Check Point 2

Erase the flash memory with referring to "5.3.8 Erase microcontroller built-in flash memory".

4.2.4 "Disabling the on-chip debug function is prohibited." (F0c33)

Basically, this error occurs when you start (download) the debugger without doing the settings described at "2.4 Set Linker Options". Do the same checking processes as"4.2.3 The on-chip debug function had been disabled in the device. (F0c79)".

CHAPTER 5　Other Information

This chapter explains some useful operation techniques of development tools and circuit diagram of the kit for developing of user programs.

6.1 Create a new workspace (project)

6.2 Register additional source file

6.3 Debugger tips

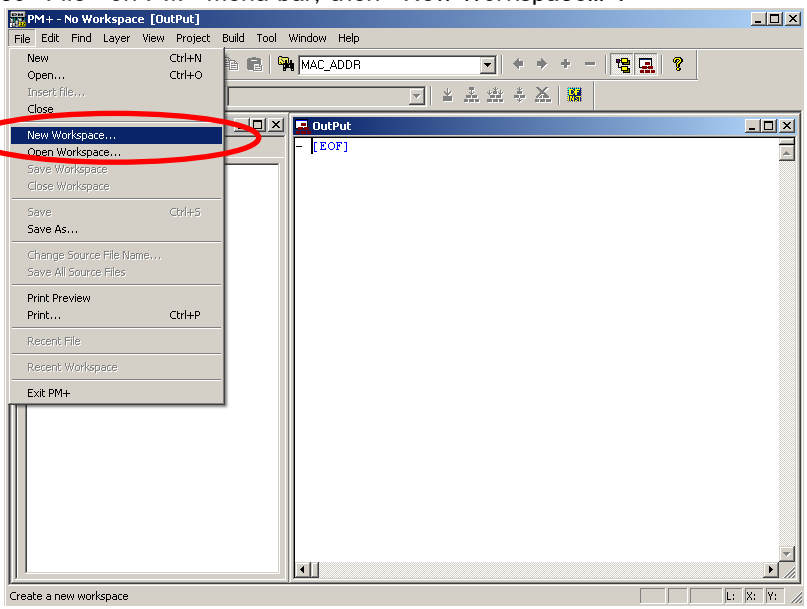
6.4 Circuit diagram

5.1 Create a new workspace

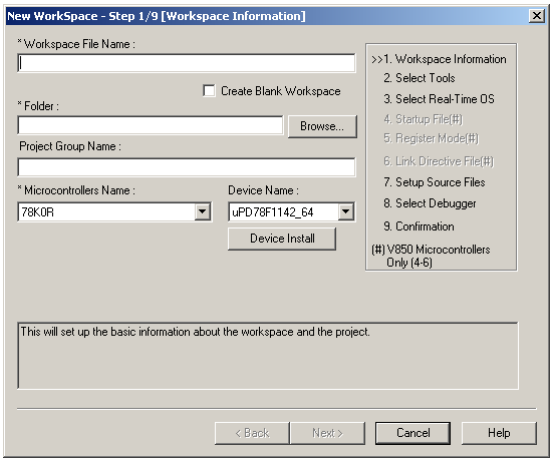
Now, create a new workspace and project.

PM+ allows you to create a new workspace with following "New WorkSpace" dialog.

Select "File" on PM+ menu bar, then "New Workspace...".



"New WorkSpace" dialog opens



<Description of items>

Workspace File Name:
-> Specify the name of the workspace file that manages the project files.
.prw is automatically suffixed as the file type.
A project file (.prj) of the same name is simultaneously created.

Folder:
-> Specify the folder for saving the workspace file by writing its absolute path.
This item can be selected from a reference dialog box by pressing the Browse... button.

Project Group Name:
-> Specify this item if wishing to manage multiple projects together in function units.
If nothing is specified, this item is the same as the workspace file name.

Microcontroller Name:
-> Specify the name of the microcontroller to be used.

Device Name:
-> Specify the name of the device to be used.

The concrete information set here is described on the following pages

Input the workspace information setting as follows.

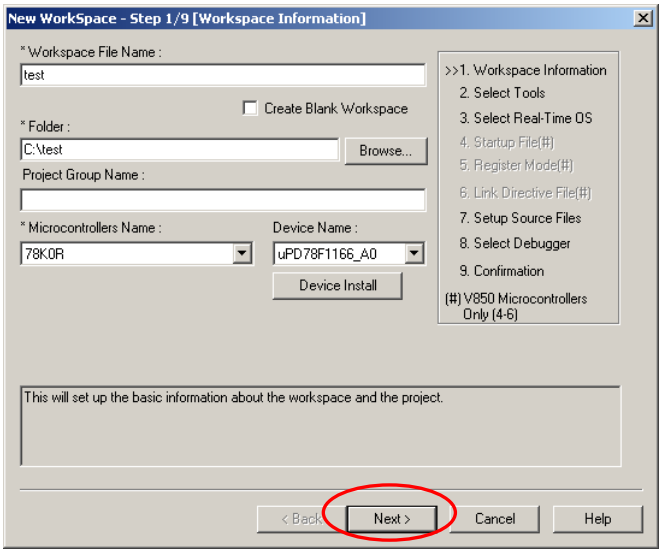
Workspace file name
→ test

Folder
→ C:¥test

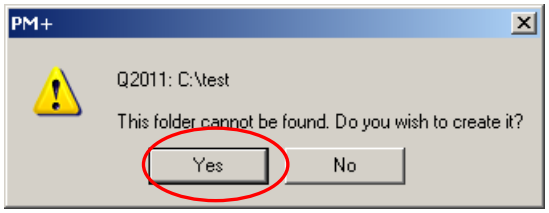
Project Group Name
→ (no input)

Microcontroller Name
→ 78K0R

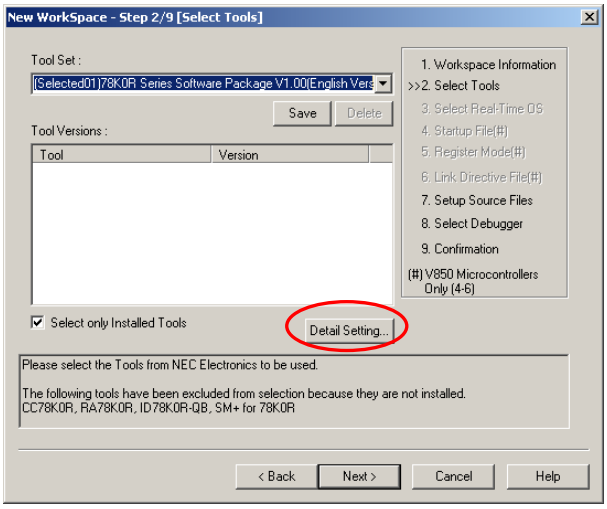
Device Name
→ uPD78F1166_A0



Click Next > button

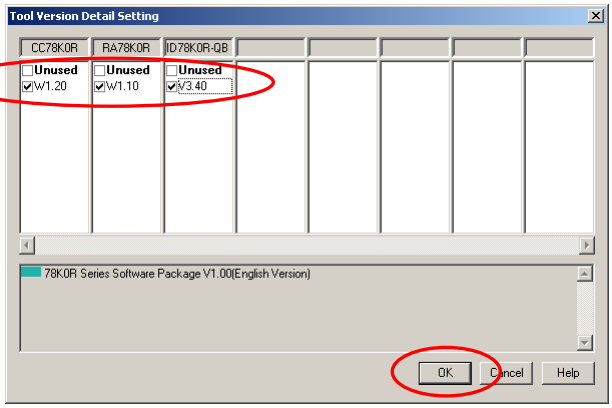


Click Yes button

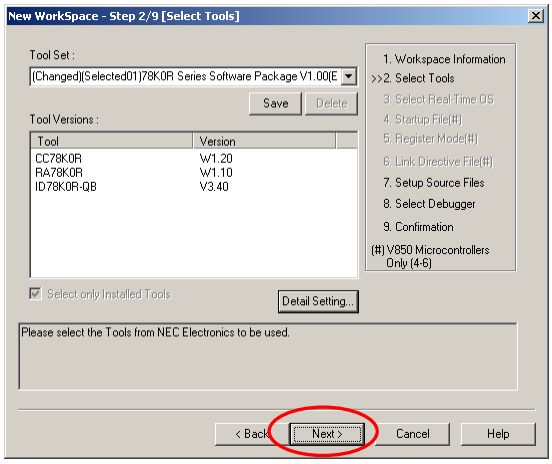


Click Detail Setting button

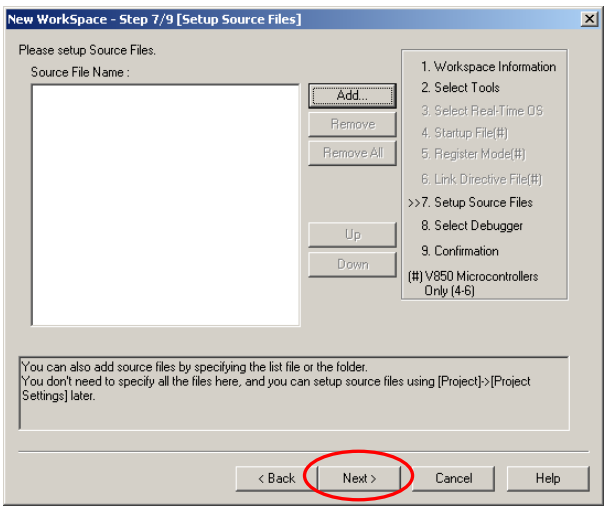
Set the version of tools as follows.
CC78K0R : W1.20
RA78K0R : W1.10
ID78K0R-QB : V3.40



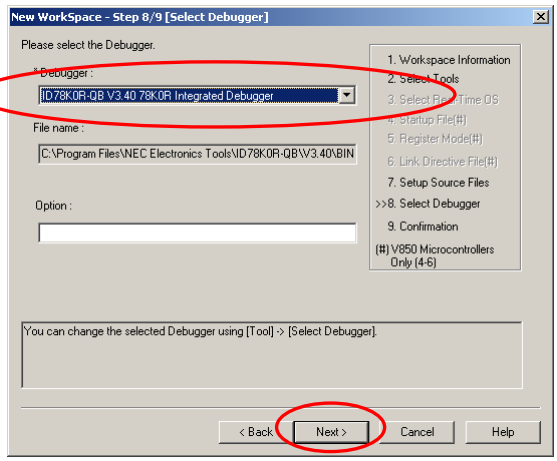
Select tools as above screenshot, then click OK .



Click Next >

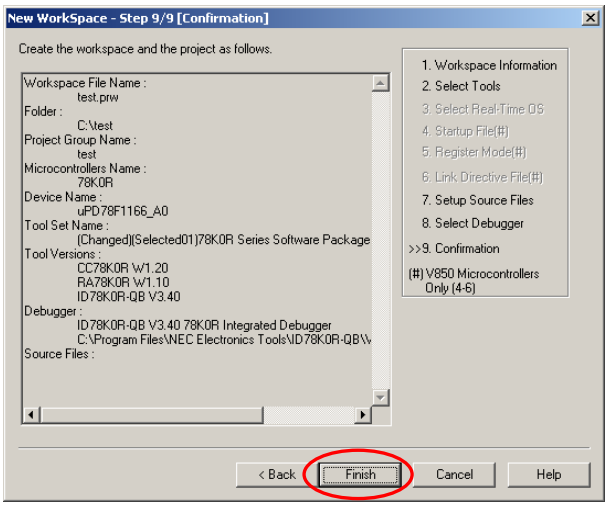


Click Next >



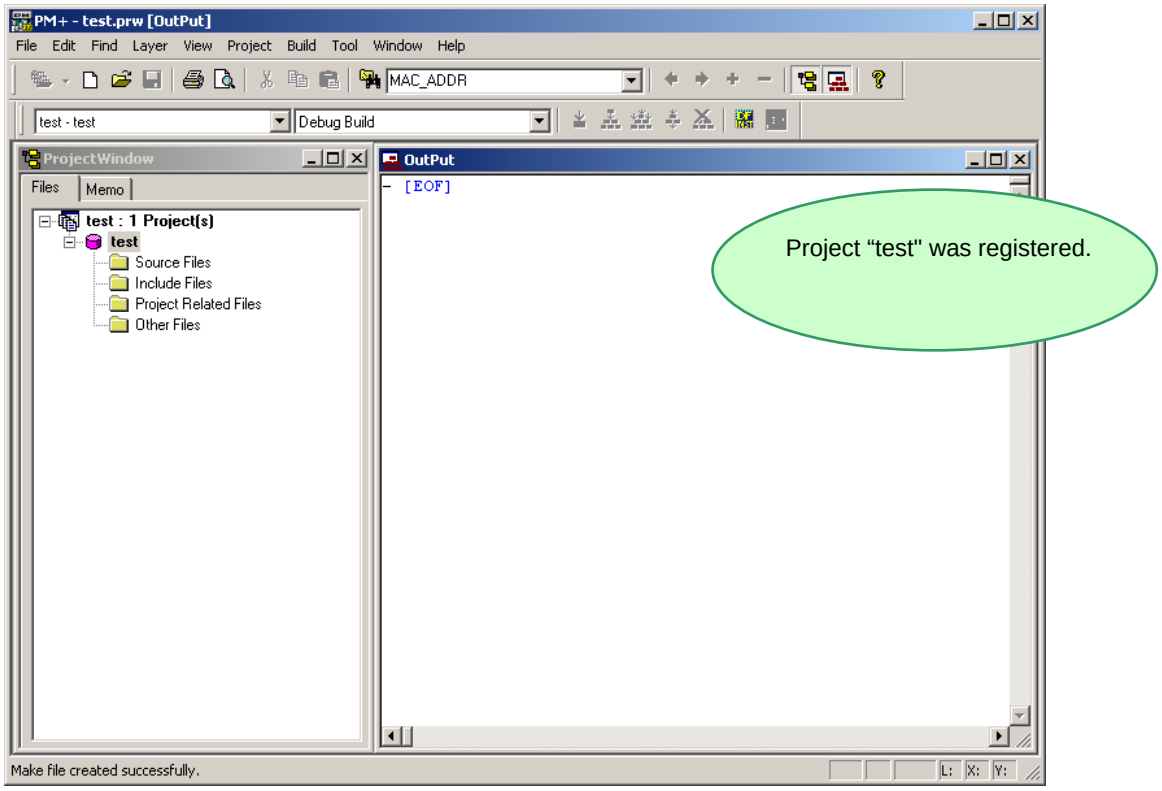
Select ID78K0R-QB V3.40

Click Next >



Check the project information settings

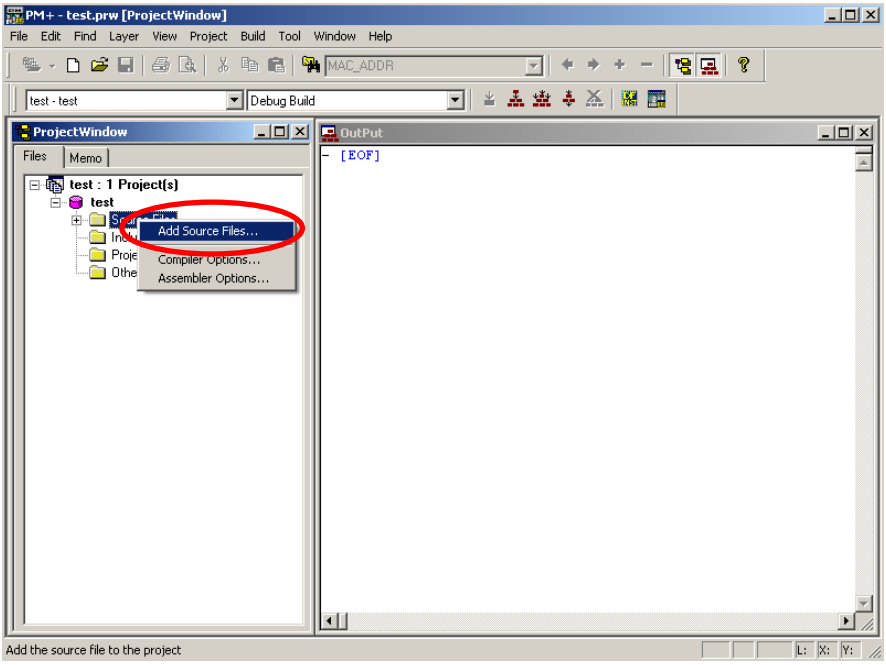
Click Finish



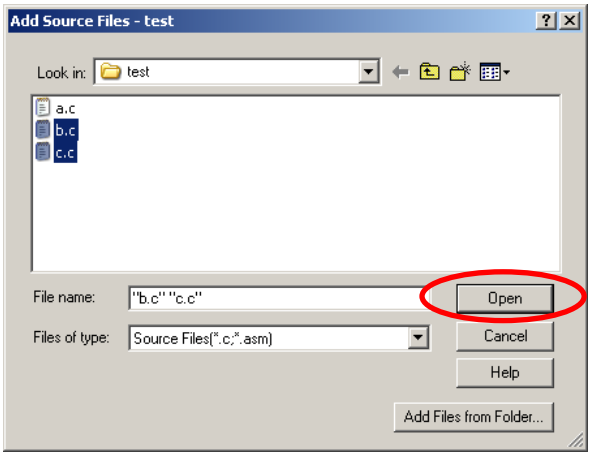
This completes workspace and project creation.
Additional source files can be registered at any time thereafter.
For details, refer to "Register additional source file".
Also, you need to do the settings for on-chip debug. Please refer to "2.4 Set Linker Options", "2.5 Set Compiler Options", and "2.7 Check Debugger Settings".

5.2 Register additional source file

Now, register additional source files.
The following example shows the additional registration of source files "b.c" and "c.c" with source file "a.c" already registered.
Place the cursor on the source file in the Project window of PM+, and select [Add Source Files…] displayed in the right-click menu.

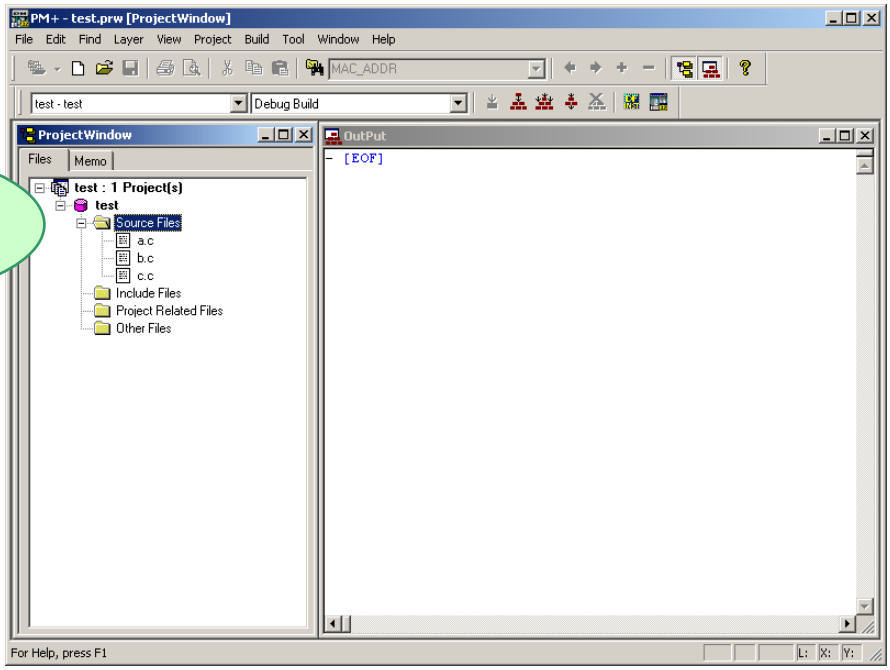


Select source files "b.c" and "c.c", then click Open



Multiple source files can be selected by clicking them with pressing Ctrl key.

Source file "b.c" and "c.c" are additionally registered to the project.



5.3 Debugger tips

This section describes some useful techniques for the debugger (ID78K0R-QB).

5.3.1 Change display of buttons

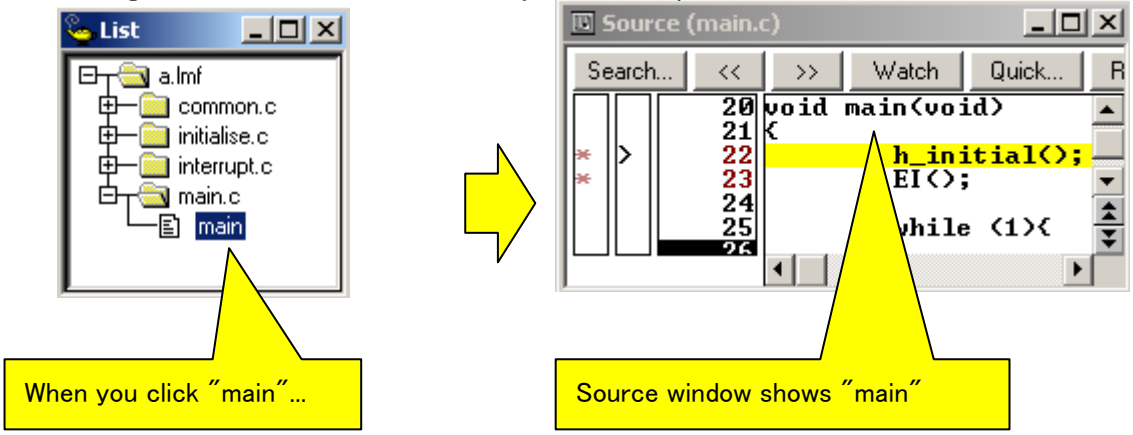
Execution controls (run, stop, step-in debugging, reset, etc) and opening functional window can be made by below buttons. However, it could be difficult to know which button does what.

In this case, select "Options" on menu bar, then "Debugger Options". Check "Pictures and Text" on setting area.

With this setting, the buttons display the text as well, so that it is easier to know what they are.

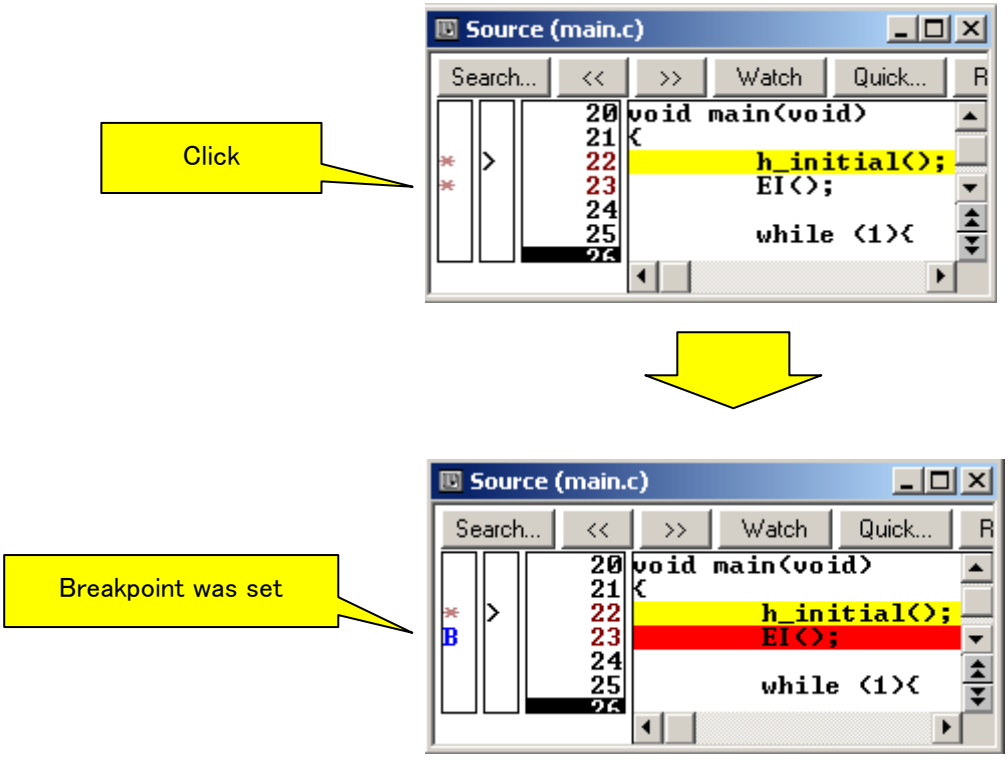
5.3.2 Display source list and function list

When you wish to see source file list or function list, select "Browse" on menu bar, then "Other" -> "List" to open the list window. The information in the windows is synchronized. Therefore, it is not just for referring to the list, but it is useful when you wish to update files or functions.



5.3.3 Set/delete breakpoints

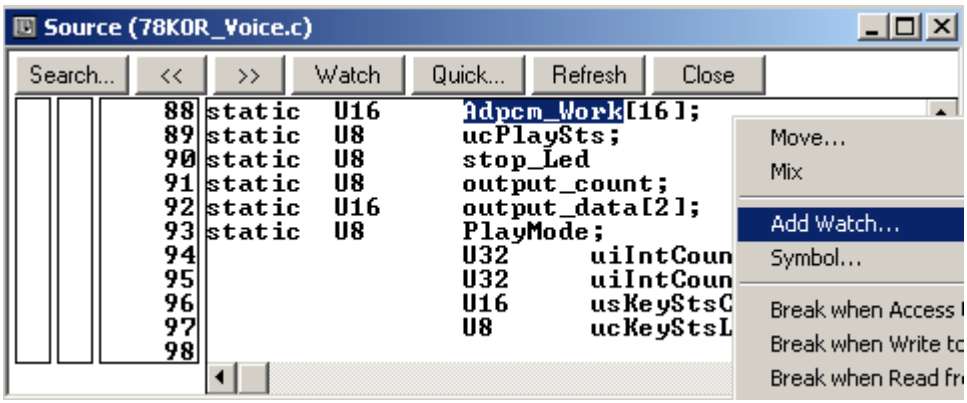
Breakpoints are executed by clicking lines in which " * " is displayed
"B" is displayed in the line where a breakpoint is set.
Breakpoints are deleted by clicking "B".



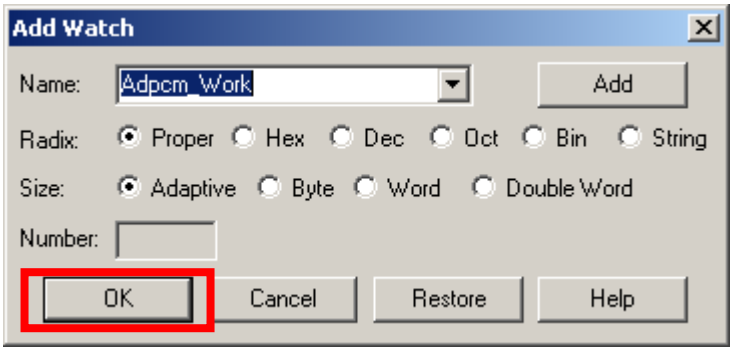
5.3.4 Display global variables

With using Watch Window, you can display global variables. There are several ways to register global variables to watch window. In this section, how to register from source window is described.

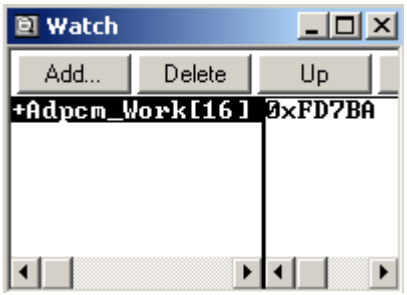
①Right-click the variable on source window, then select "Add Watch..."



②Add Watch dialog opens. Click OK .



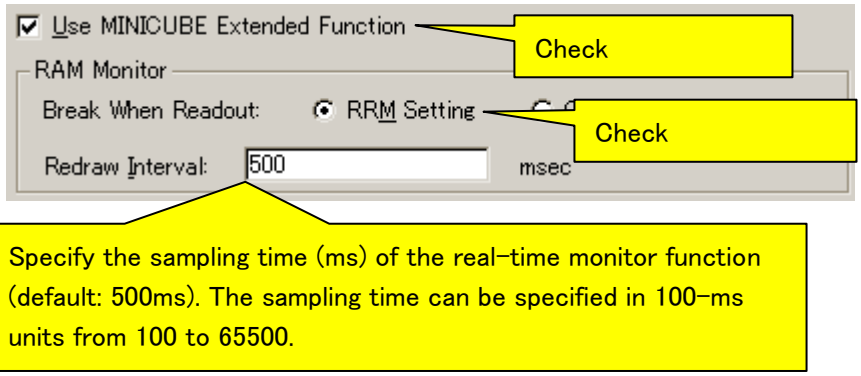
③Adding a variable to watch window is completed.



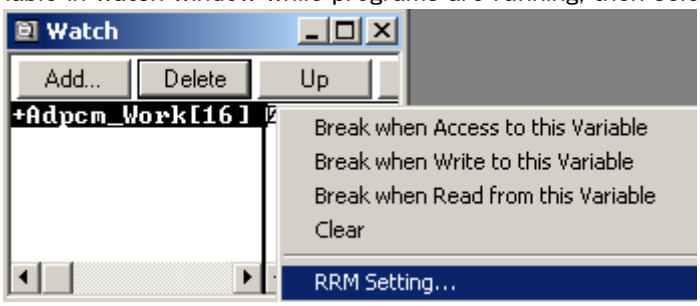
5.3.5 Display global variables while programs are running

Global variable could be referred even when the programs are running.

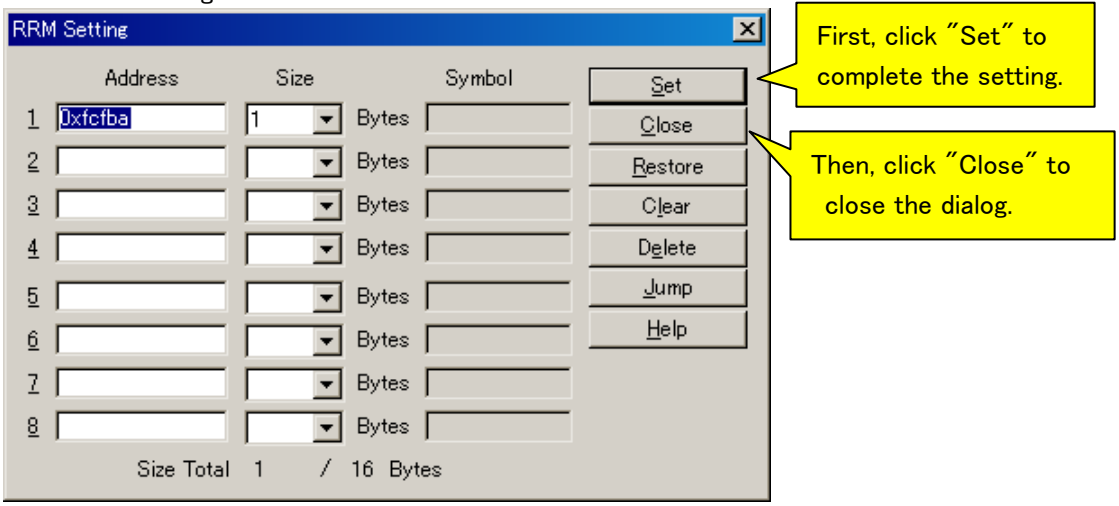
①Select "Option" menu -> "Extended Option...". Follow below settings.



②Right-click the variable in watch window while programs are running, then select "RRM Setting...".



③RRM Setting dialog opens. Click [Set] button to complete the setting, then [Close] button to close the dialog.



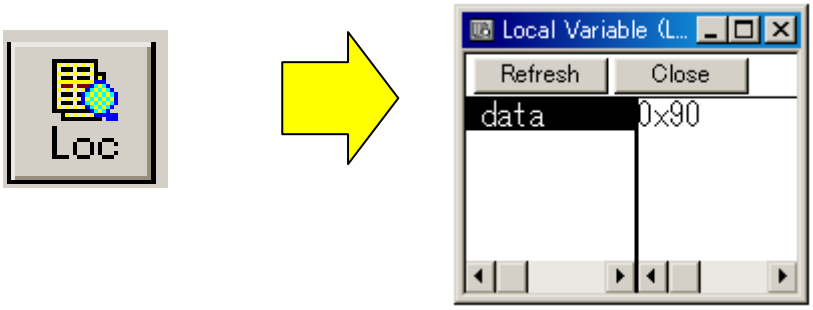
This completes the settings.

CAUTION:

- The maximum size of variable area is as total of 16byte when programs are running.
- The maximum number of variable area is 8 locations
- The user program momentarily breaks upon a read.

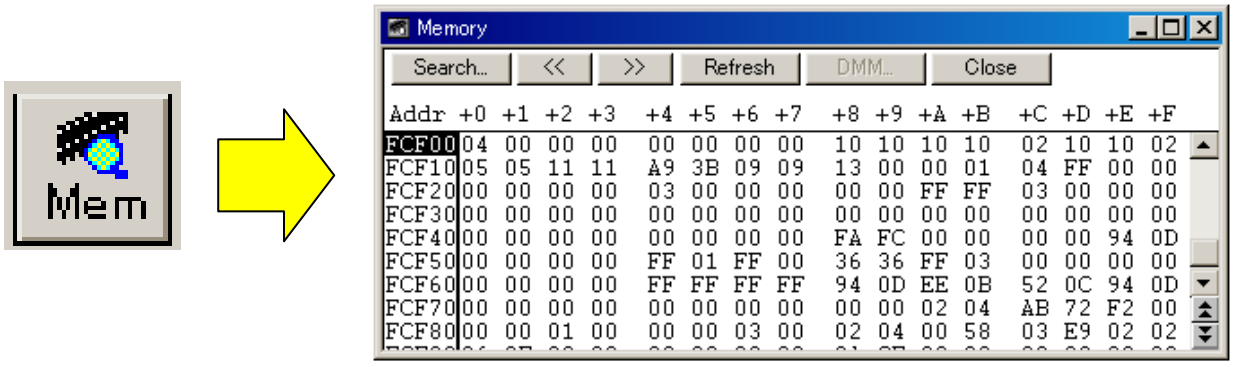
5.3.6 Display local variables

Local variable window is used to display local variables.
By clicking the button below, you can open the local variable window.
Unlike global variables, local variables cannot be displayed when programs are running.

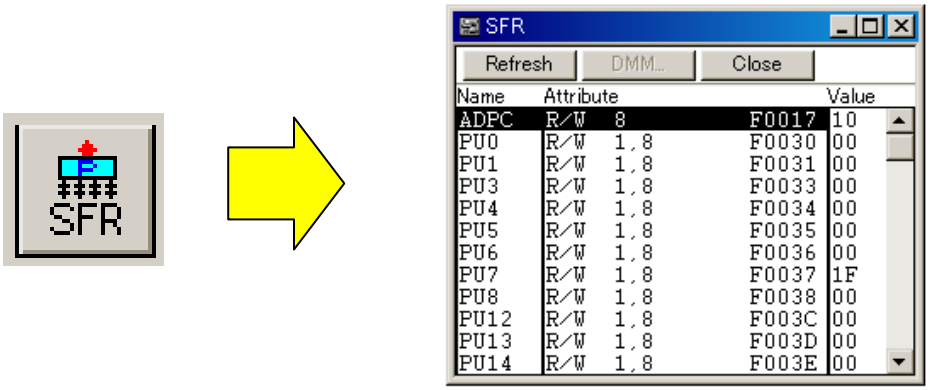


5.3.7 Display memory and SFR contents

By clicking the button below, you can open the memory window.



By clicking the button below, you can open the SFR window.

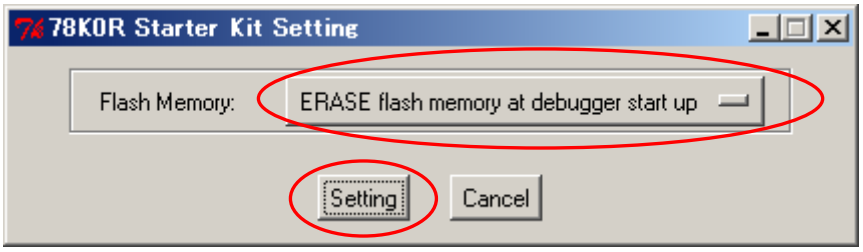


5.3.8 Erase microcontroller built-in flash memory

If you forgot the security ID or if you set On-Chip Debug Option Byte to disable the on-chip debug function, you cannot start debugger. For the case like this, there is the function to erase the flash memory.

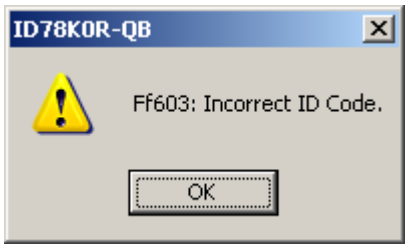
①First, check if you actually set the correct On-Chip Debug Option byte with referring to "2.4 Set Linker Options". If it is not correct, then set correctly. Remember the security ID you set.

②Use Windows Explorer to go to the path "C:¥Program Files¥NEC Electronics Tools¥TK-driver", then open "exk0r32ocfg.exe". (the directory path is for the default installation directory)
The "78K0R Starter Kit Setting" starts. Select "Erase flash memory at debugger start up" at Flash Memory, then click [Setting] .



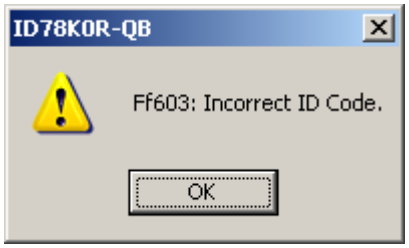
③Now, start the debugger. As you erased the flash memory with above processes, you need to enter "FFFFFFFFFFFFFFFFFFFF" (F x 20) in ID code on the configuration window of the debugger.

If you start the debugger with using PM+ debug button, the below error dialog may displayed because the debugger remember the ID Code from previous use. In this case, click [OK] to close the dialog, open the configuration dialog to enter "FFFFFFFFFFFFFFFFFFFF" (F x 20) in ID code, and then re-start the debugger again.



④When you could confirm the debugger is working, open "exk0r32ocfg.exe" again. This time, select "KEEP flash memory at debugger start up", then click Setting .
(Because there is a limit of erasing times for flash memory, it is recommended to try not to erase flash memory many times)

⑤After the debugger is started and programs are downloaded, the security ID on microcontroller changes to the code that you set at "2.4 Set Linker Options". On the other hand, the debugger stores the security ID "FFFFFFFFFFFFFFFFFFFF" (F x 20). Therefore, when you close and start the debugger again, you may get below error.

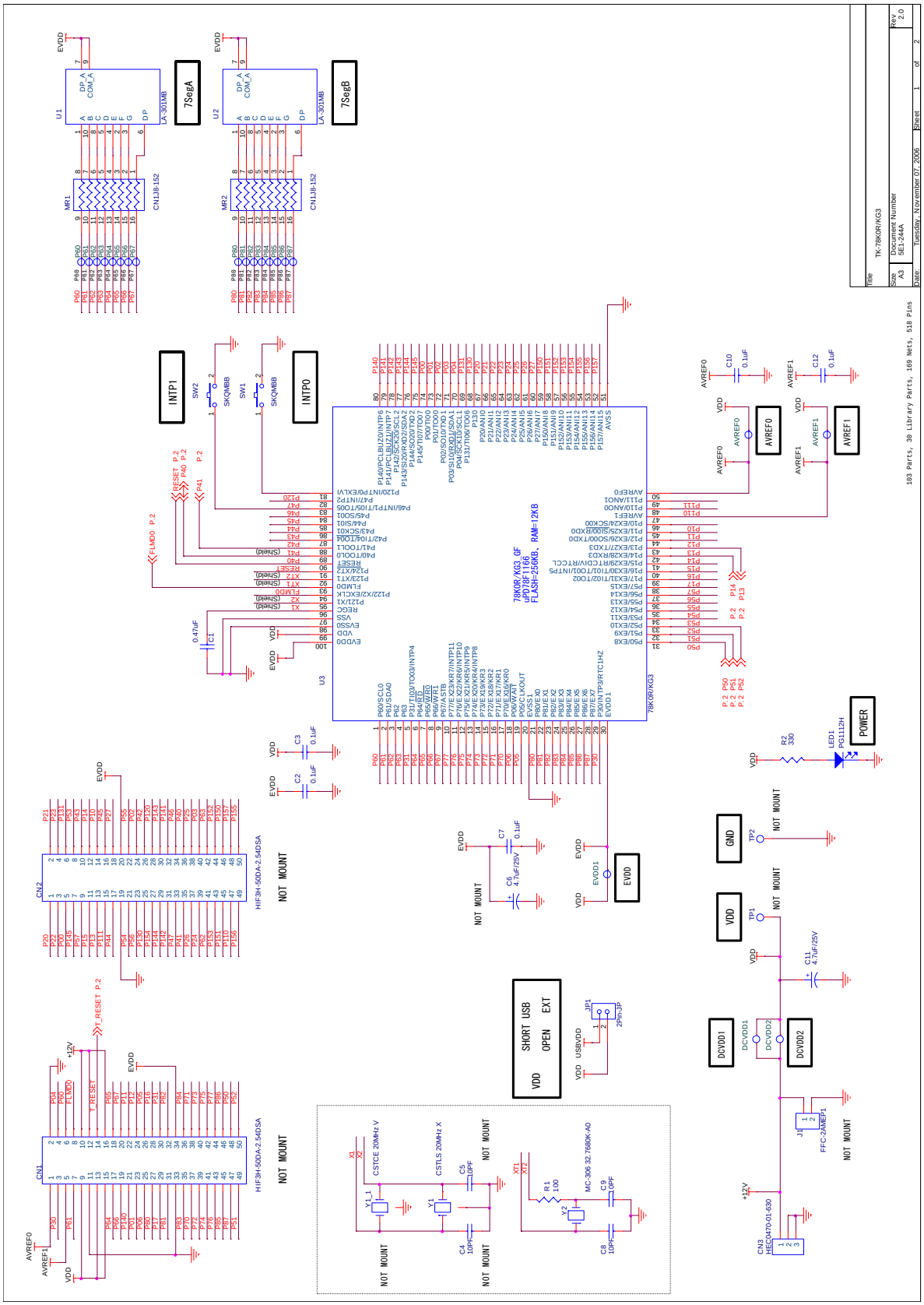


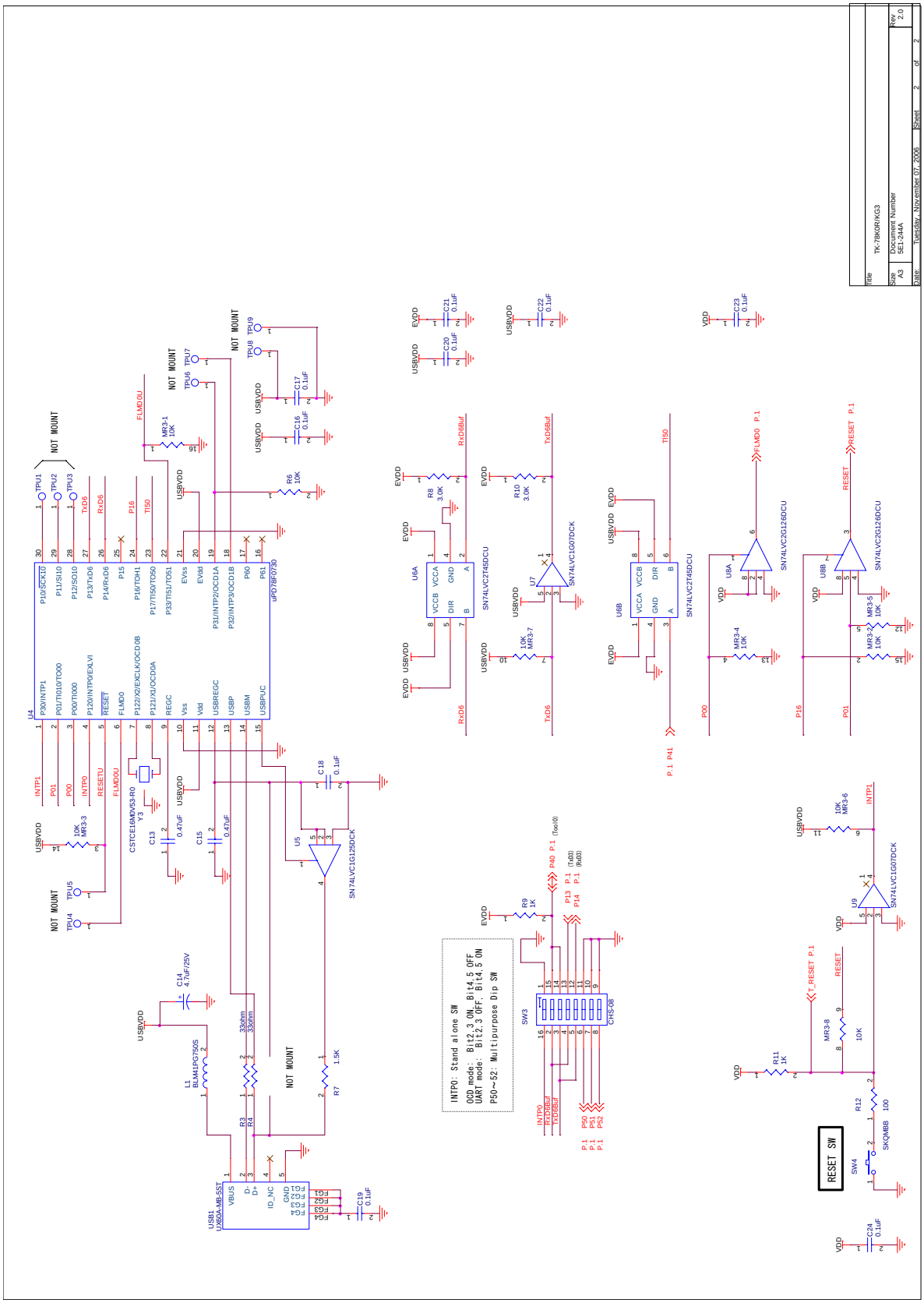
In this case, click OK to close the dialog, then enter the code you entered at "2.4 Set Linker Options" to "ID Code" on the configuration dialog.

For other information, refer to the user's manual "ID78K0R-QB Integrated Debugger -Operation-".

5.4 Circuit diagram

From following page, it shows the circuit diagram of the demonstration kit.




Title
TK-78K0R/KG3
Size
A3
Document Number
SEL2444A
Rev
2.0
Date:
Tuesday, November 07, 2006
Sheet
2
of
2
NOT MOUNT
uPD78F0730
SN74LVC2G126DCU
SN74LVC2T45DCU
SN74LVC1G125DCK
SN74LVC1G07DCK
CHS-08
RESET SW
INTP0: Stand alone SW
OCD mode: Bit2,3 ON, Bit4,5 OFF
UART mode: Bit2,3 OFF, Bit4,5 ON
P50~52: Multipurpose Dip SW